NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

TOMAHAWK STRIKE COORDINATOR PREDESIGNATION:
OPTIMIZING FIRING PLATFORM AND WEAPON ALLOCATION

by

Brian D. Kirk

September 1999

Thesis Advisor: Alexandra M. Newman

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. AGENCY USE ONLY (Leave Blank)	**2. REPORT DATE** September 1999	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE Tomahawk Strike Coordinator Predesignation: Optimizing Firing Platform and Weapon Allocation	**5. FUNDING NUMBERS**
6. AUTHOR(S) Kirk, Brian D.	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Dahlgren Division Naval Surface Warfare Center Combat Systems Branch B35 Dahlgren, VA 22448-5100	**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.	**12b. DISTRIBUTION CODE**

13. ABSTRACT ***(Maximum 200 words)***

Ships, submarines, and missiles are currently manually allocated for naval strike warfare tasking. Naval Surface Warfare Center Dahlgren Division has proposed to the Office of Naval Research to develop automated "predesignation" aids that automatically allocate the Tomahawk Land Attack Missile, at both the Tomahawk Strike Coordinator level, and at the individual firing platform level. A mixed integer program is introduced for Tomahawk Strike Coordinator predesignation, and is implemented in alternate forms and tested for a variety of scenarios.

14. SUBJECT TERMS Tomahawk Land Attack Missile, Mixed Integer Program, Weapons Allocation	**15. NUMBER OF PAGES** 92
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239.18

THIS PAGE INTENTIONALLY LEFT BLANK

TOMAHAWK STRIKE COORDINATOR PREDESIGNATION: OPTIMIZING FIRING PLATFORM AND WEAPON ALLOCATION

Brian D. Kirk
Lieutenant, United States Navy
B.S., Maine Maritime Academy, 1993

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1999

Author: ______________________
Brian D. Kirk

Approved by: Alexandra M. Newman
Alexandra M. Newman, Thesis Advisor

____
Gerald G. Brown, Second Reader

Richard E. Rosenthal
Richard E. Rosenthal, Chairman
Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Ships, submarines, and missiles are currently manually allocated for naval strike warfare tasking. Naval Surface Warfare Center Dahlgren Division has proposed to the Office of Naval Research to develop automated "predesignation" aids that automatically allocate the Tomahawk Land Attack Missile, at both the Tomahawk Strike Coordinator level, and at the individual firing platform level. A mixed integer program is introduced for Tomahawk Strike Coordinator predesignation, and is implemented in alternate forms and tested for a variety of scenarios.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TERMS AND ABBREVIATIONS

Back-up missile	A missile, associated with a *primary missile* on another ship or submarine, that a firing platform is directed to align and prepare for launch. If the firing platform that is tasked to launch the *primary missile* experiences a failure of that missile, the Tomahawk Strike Coordinator (TSC) will order the platform with the *back-up missile* to launch.
Brilliant Anti-Tank round (BAT)	A weapon designed to locate and destroy tanks.
Block	A package of improvements to a missile. There are currently two *blocks* of Tomahawks in inventory, Block III and Block II. Block III improvements include Global Positioning System (HRS) navigation, lighter warheads for C variants, and enhanced inertial navigation.
Commander In Chief (CINC)	The officer in command of a theater of operations.
Expanded Missile Identification (XMID)	A series of numbers that provide detailed information about a missile's software, including the version of operational flight software (OFS), and the version of HRS flight software (GFS). The *XMID* is used in determining position on the *M^3* list.
First Pre-Planned Waypoint (FPPWP)	The starting point of a Tomahawks over-land flight path, included in the mission data. Firing platforms are responsible for planning the over-water portion of the flight path: the flight from the firing platform to the *FPPWP*.
GREEN CROWN	A surface ship responsible for the air defense of an Amphibious Readiness Group.

Half-module	A component of the Mark 41 *Vertical Launch System (VLS)* consisting of four individual cells. Each cell provides storage and launch preparation for a missile in a canister. Due to missile power requirements and *VLS* design, only one missile per half-module may be prepared for launch at a time.
Human Computer Interface (HCI)	A tool that allows an operator to interact with a computer utility.
INDIGO	A standard Navy message that directs a firing platform to align, prepare, and launch Tomahawks. *INDIGO* assigns *primary*, *ready-spare*, and *back-up* missiles. It may direct which cells and missiles are to be used, or give the number of missiles for each task, leaving cell and missile selection to the firing platform.
Launch Sequence Plan (LSP)	A message similar to *INDIGO*, but not granting authority to align, prepare, or launch missiles. It is generally issued in advance of an *INDIGO* as a planning aid for the firing platforms.
Missile Mission Matching (M^3)	A list associated with each *mission ID* that contains all of the missile types that may be fired for that mission. The list is ordered in ascending missile capability. To maintain the greatest capability for follow-on tasking, it is desirable to select a missile from as close to the top of the M^3 list as possible.
Mission Distribution System (MDS)	A tactical computer used for managing database information about firing platforms, missions and weapons, and for displaying *TLAM* mission planning information.
Mission ID Number (mission ID)	A number corresponding to a single, unique, over-land flight path and target aimpoint.

National Command Authority (NCA)	The members of the executive branch of government authorized to direct the use of military force.
Naval Surface Warfare Center Dahlgren Division (NSWCDD)	A naval command tasked with, among other things, conducting research on weapon system development and use.
Primary missile	A missile that a firing platform is directed to align, prepare for launch, and fire.
Ready-spare missile	A missile, associated with a *primary missile* on the same firing platform, that a firing platform is directed to align and prepare for launch. The firing platform will not fire the *ready-spare* missile unless its associated *primary* fails to launch, or fails to transition from the boost phase to the cruise phase of flight. If the *primary missile* fails, the *ready-spare* will be launched without further direction from the *Tomahawk Strike Coordinator.*
RED CROWN	A surface ship responsible for the air defense of a Carrier Battle Group.
Tactical Tomahawk Weapons Control System (TTWCS)	Proposed upgrade to the shipboard systems used for Tomahawk engagement planning.
Time on Top (TOT)	The desired time to have the tasked Tomahawk impact the aimpoint.
Tomahawk Land Attack Missile (TLAM)	A cruise missile, fired from surface ships and submarines, capable of striking land targets.
Tomahawk Strike Coordinator (TSC)	The officer on the Battle Group or Naval Component Commander staff responsible for the employment of Tomahawk missiles by the Battle Group.
Variant	The type of warhead on a missile. "C" missiles have a unitary warhead; "D"

	missiles have multiple triple-effect bomblets.
Vertical Launch System (VLS)	The storage, preparation, and launch system for several types of weapons, including *TLAM*.

ACKNOWLEDGMENT

The author would like to acknowledge the financial support of Naval Surface Warfare Center Dahlgren Division. The assistance of Charles Fennemore, Robert Taft, and Donna Oliver of Naval Surface Warfare Center Dahlgren Division was invaluable in completing this research.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Employment of the Tomahawk Land Attack Missile (TLAM) in strike warfare is a mission area that has matured greatly since its inaugural use in Operation Desert Storm. Doctrine has been developed to govern the employment of TLAM in extended warfare, or campaign operations, as well as in measured response, or contingency, operations. Surprisingly, there is no official guidance governing the complex process of allocating missiles aboard ships and submarines to meet National Command Authority (NCA) or theater Commander In Chief (CINC) tasking. The planning that is crucial to the success of current operations, and that determines the ability of the on-scene force to conduct future operations, is done manually by each individual Tomahawk Strike Coordinator (TSC) and firing platform. No Tactical Decision Aids, Naval Warfare Publications, or Tactical Memoranda exist to assist the fleet operator in producing consistent, reproducible, and logical solutions. Forward deployed Battle Group assets will continue to be the on-call response force for implementing U.S. policy. Introduction of a systematic method to meet current tasking while maximizing capability for follow-on strikes, maximizing task achievement, and minimizing the impact on other operations, offers inestimable value to both the fleet and the nation.

Naval Surface Warfare Dahlgren Division (NSWCDD) has proposed to the Office of Naval Research (ONR) to develop "predesignation" aids that will automate allocations for TLAM, for both the TSC, and the individual firing platform. In support of this effort, we examine the factors that the TSC must consider when assigning tasking, introduce an

allocation method for TSC predesignation, and explore alternatives for implementing the method.

Through informal discussions with NSWCDD and Tactical Training Group Atlantic, we have determined priorities to guide the selection of firing platforms and missiles. In descending order of importance, these priorities are:

a. Meet all assigned tasking.
b. Minimize the use of firing platforms that are already occupied in other operations.
c. Maximize the use of missiles from designated "expend" platforms. These are ships and submarines from which it is desired to use as many missiles as possible. This may occur, for example, if the firing platform is leaving the theater of operations, or is heading into an in-theater port for extensive repair work.
d. Use missiles from the platforms not designated as "expend" in such a way as to level the number of TLAM's remaining on these non-expend platforms. This is done to prevent the inventory of a particular firing platform from being exhausted, rendering the platform unable to perform future tasking.
e. If desired by the operator, spread tasking among as many firing platforms as possible to limit the chance of a single point of failure.
f. Where possible, use the least capable missile for each mission. This attempts to limit the use of missiles with specific capabilities (e.g., GPS navigation) to missions that require those capabilities.
g. Maximize follow-on firing capability.

Using these priorities, we formulate a mixed integer linear program to perform allocations. This consists of a multiple-term objective function that expresses the relative priorities, subject to a series of constraints. This constrained optimization model attempts to allocate missiles in such a way as to reflect all of the priorities, while meeting operational restrictions on the use of missiles and firing platforms.

We initially attempt to solve this mixed integer linear program as a single, monolithic problem. We have discovered, however, that the size of the problem renders this method intractable. Limits on machine precision, and the necessity to accept solutions that, while not optimal, are within a specified range of being optimal, result in solution times that are too long to be operationally useful, and are of unacceptable quality, i.e., the solutions returned do not adequately capture all of the priority levels.

We next use a hierarchical restriction method in which we express the formulation as a series of sub-problems, one for each priority level. These sub-problems are solved in descending order of priority. This method accurately captures all of the priority levels, and returns solutions of acceptable quality; however, the solution time far exceeds what could be considered tactically viable.

Our final approach is the use of a simplified heuristic, or rule-based system, to assist in rapidly finding solutions. The heuristic uses conditional checks to determine if certain sub-problems can be eliminated from consideration, uses selectively restricted data sets to solve some problems, and evaluates certain sub-problems using allocations that were obtained in previous sub-problems. The resulting solutions are of comparable quality to those obtained with the hierarchical restriction method, and drastically reduce the solution time required.

We conclude that solving TSC predesignation as a monolithic mixed integer problem is not viable. We recommend that NSWCDD develop a complete (non-integer programming-based) heuristic for this application. A signal disadvantage of a heuristic is that it is seldom possible to objectively assess the quality of its solutions. The mathematical models and formal methods introduced here may be useful in independently assessing the quality of the heuristic solution. We further recommend that, at an early stage of development, this heuristic be presented to the strike warfare community for review. The goal of this review by decision makers is to determine the suitability of the heuristic for tactical systems, and to establish doctrinal ordering of priorities. We recommend that NSWCDD advocate the acceptance of the heuristic by the Naval Doctrine Command, Fleet Commanders, and Theater CINC's, and its inclusion in the Naval Warfare Publications and Concepts of Operations that govern the conduct of strike warfare. It is crucial that strike planning be standardized in published doctrine so that it can be executed with the effectiveness that it deserves.

I. INTRODUCTION

A. BACKGROUND

The Tomahawk Land Attack Missile (TLAM) has proven to be the weapon of choice for many U.S. Navy strike warfare scenarios (Figure 1). Doctrine has been developed to guide the employment of TLAM in extended warfare, or campaign, operations, as well as in measured response, or contingency, operations.

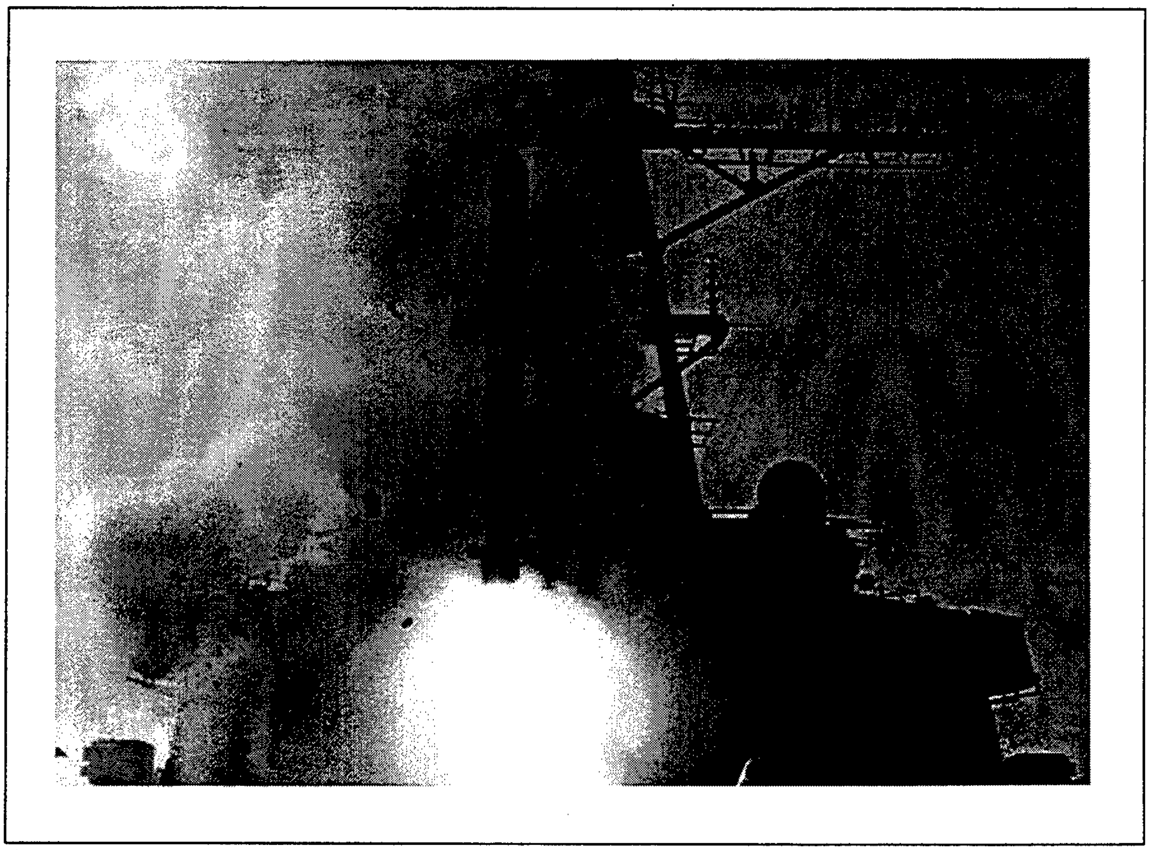


Figure 1: An ARLEIGH BURKE Class Destroyer Launches a TLAM During Operation DESERT FOX [U.S. Navy, 1999a].

Allocating individual ships, submarines, and missiles to meet National Command Authority (NCA) or theater Commander In Chief (CINC) tasking is complex. Each individual Battle Group Tomahawk Strike Coordinator (TSC) and firing platform currently uses manual methods for these allocations. There are no Tactical Decision Aids, Naval Warfare Publications, or Tactical Memoranda to assist in producing a solution. No systematic way currently exists to meet current tasking while maximizing capability for follow-on strikes, maximizing task achievement, or representing the difference between the assignment of TLAM tasking to a firing platform that is already occupied with another real-world mission rather than a firing platform that is more available.

Forward deployed Battle Group assets will continue to be the on-call response force for implementing U.S. policy. This includes not only TLAM tasking, but diverse missions such as Battle Group air defense (RED CROWN), amphibious readiness group air defense (GREEN CROWN), non-combat evacuation of American citizens overseas, enforcement and monitoring of no-fly zones, humanitarian missions, maritime interdiction, and coalition and treaty operations and exercises. As the demands on operational forces increase, fuel allocations remain constant or decrease, and weapons costs increase, it becomes more vital to use each ship and missile in the most effective manner.

With the development of improved strike munitions, more options will be available for the NCA and CINC, such as Brilliant Anti-Tank (BAT) weapons and Tactical Tomahawk, allowing for rapid "call for fire" strike missions as well as the traditional attack against a known, stationary target. This will add complexity to weapon and ship allocation to meet tasking while reducing the amount of time available for strike planning, making manual allocation methods even less viable.

Automated selection algorithms for the allocation of firing platforms for TLAM and follow-on strike munitions is desirable to provide consistent, logical, and reproducible solutions. Consistency is necessary because tasking may be performed at several levels. The TSC, or higher authority, may task to the firing platform level, leaving the individual firing platforms to select missiles and cells or torpedo tubes. In a more detailed scenario, the tasking may be specified to a particular missile. Any system of allocation should produce the same results, whether it is run in the Combat Information Center of a destroyer, or on the aircraft carrier where the Battle Group staff is embarked. Consistency also allows for a smooth transition between the TSC and an alternate commander in the event that it becomes necessary to shift TSC duties. Logical solutions are needed to ensure that the weapon selection follows doctrine. Reproducible solutions promote effective training, and allow intelligent discrimination between alternate scenarios. None of these desirable characteristics exist with the current method in which each operator manually selects firing platforms and missiles, based on his or her experience and intuition. The lack of a uniform method for assigning assets to meet tasking denies the fleet the ability to maximize remaining power projection capability, ensure optimal tactical employment, and meet tasking with the least capable missile. A fleet-wide system of automated TLAM "predesignation," is one way to meet all of these criteria.

B. PAST WORK

Naval Surface Warfare Center Dahlgren Division (NSWCDD) has formulated and implemented a mixed integer program that selects missiles to meet assigned tasking for an individual surface ship [Naval Surface Warfare Center Dahlgren Division, 1997]. This single-ship predesignation model is designed to be incorporated in the TLAM fire control system. The Naval Postgraduate School (NPS) thesis "Optimizing Selection of Tomahawk Cruise Missiles" by LT Scott Kuykendall presents an alternate formulation that is also capable of performing single-platform allocation for a surface ship [Kuykendall, 1998]. Kuykendall's mixed integer program additionally incorporates submarine allocation, and is extendible to a multiple firing platform Battle Group.

C. PRESENT WORK

Based on the success of both the NSWCDD and NPS formulations, NSWCDD has submitted a proposal for a multi-year project to the Office of Naval Research (ONR) [Fennemore, 1999c]. The goal of this project is to provide the fleet with a mathematically defensible solution to the problem of which specific firing platforms and missiles to use to meet strike tasking. Fleet users are considered both at the TSC level, and at the individual firing platform level. Due to differences in weapon system configurations, surface ships and submarines are considered separately.

The predesignation system should consist of three separate utilities. TSC predesignation is designed to be embedded in the Mission Display System (MDS), and is intended for use by the Battle Group TSC to automate the selection of firing platforms, number of missiles for each platform, and, if desired, individual cells or torpedo tubes in order to optimize missile selection for the entire Battle Group. The output from the TSC solver can then be used to write LSP's or INDIGO's (formatted messages which provide advance planning information and authority to launch TLAM's, respectively). Surface ship predesignation is designed to be included in the Tactical Tomahawk Weapon Control System (TTWCS), and used when the TSC does not provide tasking for specific cell selection. Surface ship predesignation will take an INDIGO or LSP and own-ship information as input, and provide a solution that optimizes half-module and cell selection at the individual firing platform level. Submarine predesignation is designed to be incorporated into a submarine's fire control system for use in a manner similar to surface ship predesignation, and will automate selection of individual cells or missile and torpedo tube combinations.

A brief overview of surface ship and submarine predesignation is given in this thesis to provide the reader with a more complete understanding of the entire predesignation project; the systems themselves are being developed by NSWCDD [1997, 1999].

TSC predesignation will incorporate three distinct packages: the TSC solver, the surface ship solver, and the submarine solver. The surface ship and submarine solvers will be copies of the solvers found on the individual firing platforms. The TSC solver will consist of three components: a Human Computer Interface (HCI), a pre-processor, and a solver. The HCI will allow the input and viewing of the following required data:

a. List of targets corresponding to TLAM mission identification numbers (mission ID's);
b. Number of primary missiles to use for each mission ID;
c. Number of ready-spare and back-up missiles to prepare for each mission ID;

d. Missile Mission Matching (M^3) list for each mission ID; this list specifies which weapon types may be used for a particular mission, and the relative desirability of each allowed weapon type for that mission;
e. Time on Top (TOT) for each mission ID;
f. Firing platform geographic location;
g. Firing platform current tasking, e.g., maritime interdiction operations;
h. Firing platform missile load-out data, i.e., missile block, variant, and Expanded Missile Identification (XMID) information down to the cell/torpedo room level, for each firing platform;
i. Firing platform mission load-out data, i.e., list of all TLAM mission ID's for which a firing platform has mission data;
j. Firing platform time remaining in theater, i.e., the amount of time a particular ship or submarine has left in the area of operations; this is significant when an individual firing platform will be departing the theater earlier than the rest of the Battle Group; and
k. The relative value of each weapon type.

[Fennemore, 1998a]

The operator will be able to specify tactical options, consisting of:

a. Individual platforms or a group of ships to use;
b. Of the platforms to be used, platforms from which it is desired to expend as many missiles as possible, e.g., platforms that are leaving the theater of operations;
c. Missiles that are "reserved" for special tasking or call-for-fire missions and not to be considered in the solution;
d. Changes to the priority order of the M^3 list that may be done due to direction from higher authority to use a missile for a task that does not require some of the capabilities of that missile; for example, the TSC may be directed to use only block III missiles for a particular task that does not require block III capabilities. This may occur in a contingency scenario governed by factors necessary to the conduct of the operation that are not of immediate tactical concern to the TSC;
e. Geographic feasibility for each platform with respect to each launch area, i.e., whether or not a firing platform can attain the necessary launch position in a particular launch area in time to meet tasking;
f. Employment penalties for each platform that assess the relative value of the mission that each firing platform is currently performing. For example, it may be more desirable to leave a cruiser on RED CROWN in the Adriatic than to use it for TLAM tasking in the East Mediterranean; and

g. Launch areas in which it is desired to spread primary or back-up missiles across as many firing platforms as possible, or, conversely, launch areas in which it is desired to concentrate primary or back-up missiles on as few firing platforms as possible.

[Fennemore, 1998b]

The pre-processor will take input from the HCI and formulate a model to pass to the solver. The solver will be a software package capable of taking the pre-processed formulation and returning a solution. The solution will be processed by the HCI and displayed.

The TSC solver will check tasking assigned to each firing platform down to the cell level. This guarantees that there is a feasible solution for all firing platforms, but does not address optimality from the firing platform perspective. If the TSC desires to task down to the individual missile level, the tasking proposed by the TSC solver will be passed to the ship and submarine solvers internal to the TSC predesignation system. These solvers will return half-module and cell selection solutions for each firing platform, which can be disseminated in INDIGO or LSP messages. If the TSC does not desire to task to the individual missile level, the output from the TSC solver will not be passed to the internal ship and submarine solvers. Instead, the solution, aggregated to the firing platform level, will be sent to each firing platform as an INDIGO or LSP, where the individual surface ship and submarine solvers will be used for assignment of tasking to specific cells within half-modules.

At the individual firing platform level, the optimization problem becomes less complex. Surface ship and submarine predesignation will consist of a single component, the surface ship or submarine solver, respectively. This will include an HCI, a pre-processor, and a solver.

D. WEAPONS SYSTEMS

There are currently three classes of surface ships capable of firing TLAM; USS TICONDEROGA (CG-47) class Aegis cruisers (with the exception of the first five ships in the class), USS ARLEGIH BURKE (DDG-51) class Aegis destroyers, and USS SPRUANCE (DD-963) class destroyers.

All three ship types are equipped with the MK41 Vertical Launch System (VLS), which is the storage, preparation, and launch system for several types of weapons, including TLAM. The MK 41 VLS employs two launcher sizes. Full-size launchers have eight modules with eight cells per module, or 16 half-modules with four cells each; see Figure 2. Three cells of Module Five are used for an ordnance handling crane, yielding a full-size launcher with a total of 61 useable cells. Half-size launchers have four modules with eight cells per module, and also incorporate a crane, leaving 29 useable cells.

Full-Size Launcher										
Mod	Cell 8	Cell 7	Cell 6	Cell 5		Cell 4	Cell 3	Cell 2	Cell 1	Mod
2	Cell 1	Cell 2	Cell 3	Cell 4		Cell 5	Cell 6	Cell 7	Cell 8	1
Mod	Cell 8	Cell 7	Cell 6	Cell 5		Cell 4	Cell 3	Cell 2	Cell 1	Mod
4	Cell 1	Cell 2	Cell 3	Cell 4		Cell 5	Cell 6	Cell 7	Cell 8	3
Mod	Cell 8	Cell 7	Cell 6	Cell 5		Cell 4	Cell 3	Cell 2	Cell 1	Mod
6	Cell 1	Cell 2	Cell 3	Cell 4		Cell 5	VLS Crane			5
Mod	Cell 8	Cell 7	Cell 6	Cell 5		Cell 4	Cell 3	Cell 2	Cell 1	Mod
8	Cell 1	Cell 2	Cell 3	Cell 4		Cell 5	Cell 6	Cell 7	Cell 8	7

Figure 2: Full-Size VLS. The full-size VLS launcher has eight modules, each with eight cells. The VLS ordnance handling crane occupies cells six, seven, and eight of module five, leaving 61 useable cells. [After U.S. Navy, 1994].

Maximum theoretical salvo size is determined by the total number of half-modules on a firing platform. This is because the MK 41 VLS only allows one missile per half-module to be prepared for launch at a time. In practice, the actual maximum salvo size also depends on how missiles are loaded into the launcher. For a particular weapon type, the salvo size is the sum of half-modules that contain at least one weapon of that type. Figure 3 shows an example of maximum salvo calculation, using a hypothetical load-out.

Mod	CII	CII	CIII	CIII		CII	DIII	CIII	DIII	Mod
2	CIII	CIII	CIII	CIII		CIII	CIII	CIII	CIII	1
Mod	DIII	CII	CII	CIII		DII	DIII	CIII	CII	Mod
4	CIII	CII	CII	CIII		CIII	DIII	CIII	CIII	3
Mod	DIII	CIII	CIII	CIII		CIII	CIII	CIII	CIII	Mod
6	DII	CIII	CIII	CIII		CIII	VLS Crane			5
Mod	DIII	CII	CIII	CII		CIII	DIII	DIII	CIII	Mod
8	DIII	DIII	CIII	CIII		CII	CIII	CIII	CIII	7

Figure 3: Sample Load Plan for a Full Size Launcher. CII, CIII, DII, and DIII refer to blocks and variants TLAM. Note that a CIII missile is loaded into each half-module; this gives a maximum salvo size of 16 CIII TLAM's from this launcher.

Cruisers (Figure 4) have two full-size VLS launchers, one forward and one aft; this provides a total of 122 cells and 32 half-modules. The maximum theoretical salvo size for a cruiser, assuming at least one TLAM in each half-module, is 32.

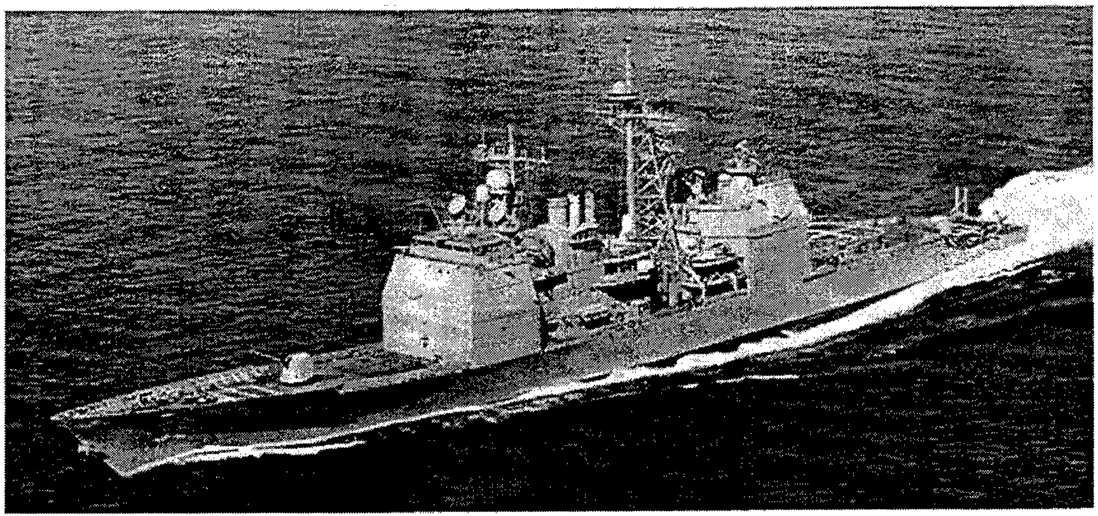


Figure 4: USS SAN JACINTO (CG-56), a TICONDEROGA Class Cruiser [U.S. Navy, 1999b]. Cruisers have two full-size VLS launchers, one forward and one aft; this provides a total of 122 cells and 32 half-modules. The maximum theoretical salvo size for a cruiser, assuming at least one TLAM in each half-module, is 32.

ARLEIGH BURKE class destroyers (Figure 5) have a half-size VLS launcher forward, and a full-size launcher aft, providing a total of 24 half-modules and 90 cells, and a maximum salvo of 24.

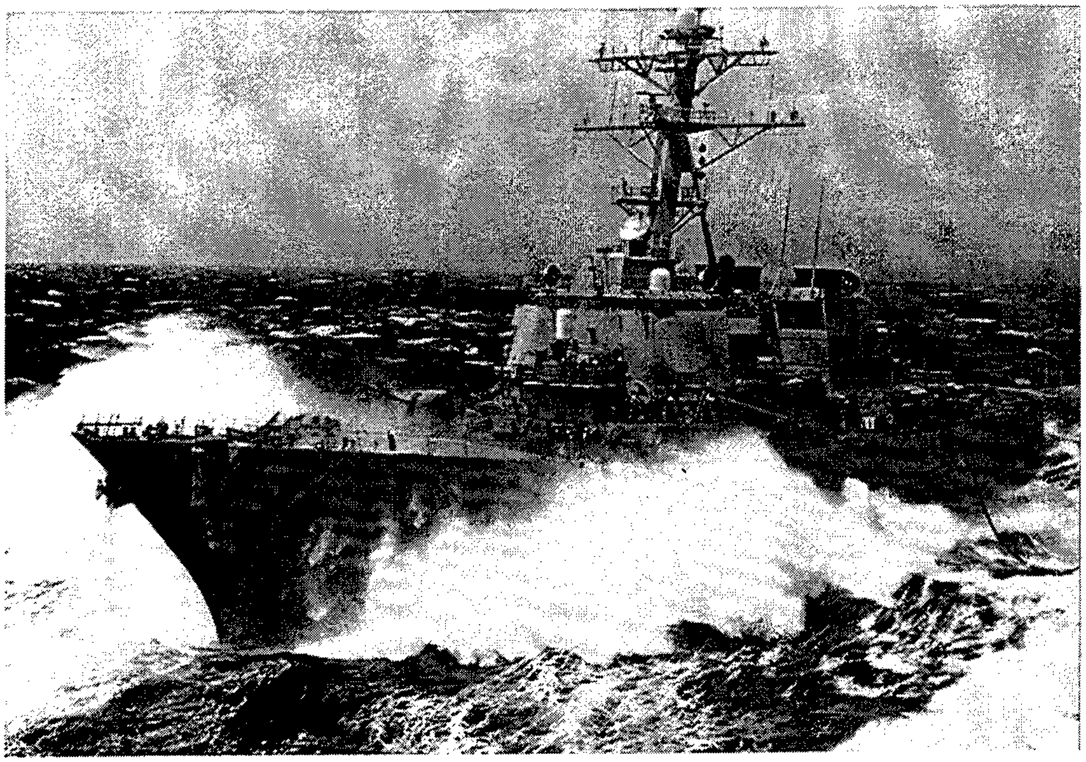


Figure 5: USS MITSCHER (DDG-57), an ARLEIGH BURKE Destroyer [U.S. Navy, 1999c]. ARLEIGH BURKE class destroyers have a half size VLS launcher forward, and a full size launcher aft, providing a total of 24 half-modules and 90 cells, and a maximum salvo of 24.

SPRUANCE class destroyers (Figure 6) have a single full-size launcher forward, 16 half-modules and 61 cells, for a maximum salvo of 16. Additionally, due to ship combat systems design, SPRUANCE class destroyers only carry TLAM's or Vertical Launch Anti-Submarine Rockets, a rocket thrown torpedo, in their VLS launchers, while cruisers and ARLEIGH BURKE class destroyers can additionally load SM-2 variant surface-to-air missiles. Therefore, in general, SPRUANCE class destroyers will be loaded out with a higher proportion of TLAM's than ships of the other two classes.

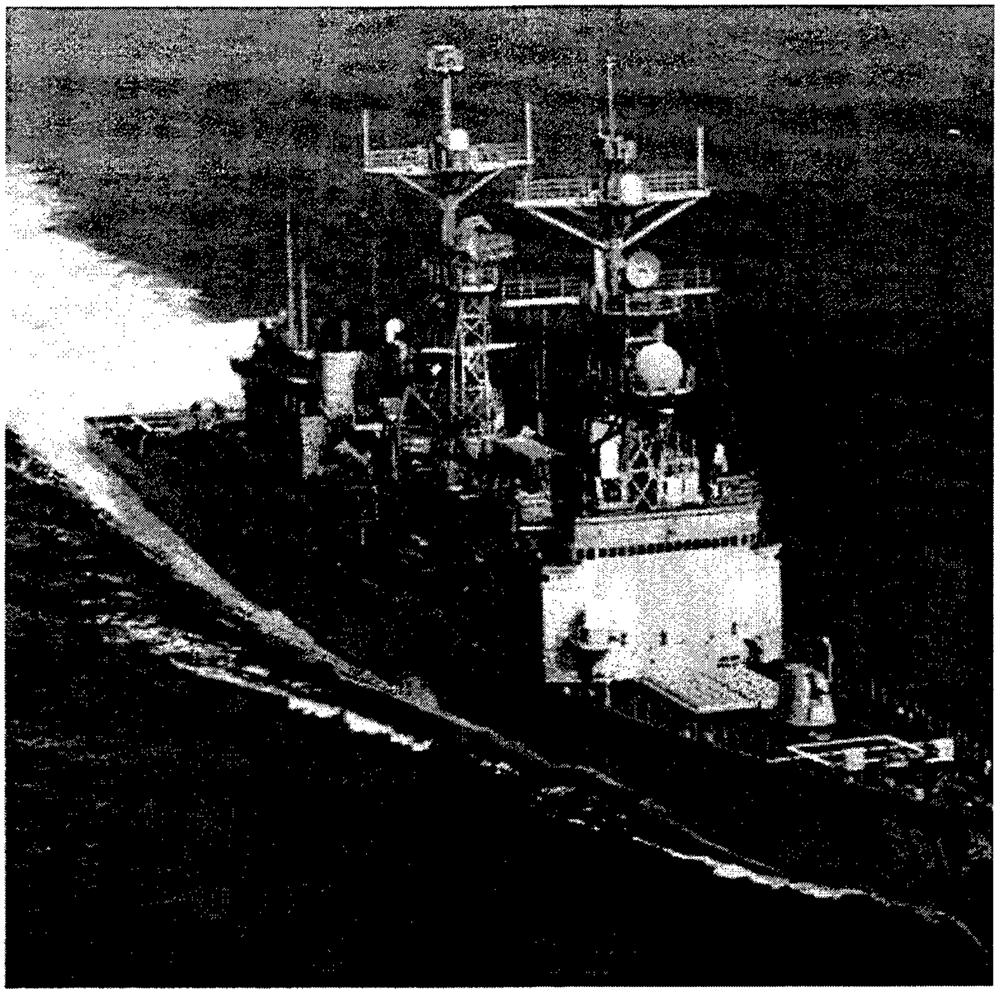


Figure 6: USS JOHN HANCOCK (DD-981), a VLS-Equipped SPRUANCE Class Destroyer [U.S. Navy, 1999d]. SPRUANCE class destroyers have a single full size launcher forward, 16 half-modules and 61 cells, for a maximum salvo of 16.

THIS PAGE INTENTIONALLY LEFT BLANK

II. MODELING APPROACHES

A. PROBLEM DEFINITION

The scope of this thesis is the formulation of the mixed integer program that will be the template for the TSC solver, and the implementation of the formulation in GAMS [GAMS, 1999]. Per discussions with NSWCDD, the formulation accounts for multiple Battle Groups, launch areas and time periods. The objective function minimizes penalties for failure to meet tasking, employment penalties, and penalties for not using the least capable missile, while maximizing follow-on salvo capability and allocating missiles across firing platforms in accordance with the tactical decisions of the TSC.

B. ASSUMPTIONS

The TSC Predesignation model is a mixed integer program with a linear objective function and linear constraints. Several assumptions and modeling approaches in the formulation need explanations.

Considerations that are unique to submarines are not addressed. NSWCDD has determined that it is sufficient to demonstrate the TSC model using only surface ships as firing platforms [Fennemore, 1999a].

An index is not explicitly defined to differentiate between firing platforms of different Battle Groups, as any Battle Group asset can be uniquely identified by its hull number. For example, USS MITSCHER has the hull number DDG-57. If USS MITSCHER is assigned to the ENTERPRISE Battle Group, referring to DDG-57 conveys the firing platform as well as the associated Battle Group.

The model treats tasking as a series of tasks and parts. Each task corresponds to a single TLAM mission ID, and consists of three parts. Parts 1, 2, and 3 correspond to the primary, ready-spare, and back-up missile, respectively. While each task corresponds to one mission ID, a mission ID may have multiple associated tasks. Each part of each task can require the firing of at most one missile. To calculate the number of missiles remaining after a strike, it is assumed that primary missiles will be launched, but ready-spare and back-up missiles will not.

Tasks are associated with launch areas. A launch area is a geographic location containing any positive number of First Pre-Planned Way Points (FPPWP's). A firing platform in the launch area can fire missiles for any mission with a FPPWP in that launch area, and cannot fire missiles for any mission outside of that launch area. This is due to the maximum ranges of the various missile types, and corresponding restrictions on the length of the over-water portion of the flight path. For example, a ship in the Adriatic launch area can fire missiles for any mission with a FPPWP in that launch area, but cannot fire any missile for missions that have FPPWP's in the North Arabian Gulf.

Missile spreading, i.e., how missiles are chosen from several operationally feasible firing platforms, depends on three features. First, it may be desirable to use as many missiles from a particular platform as possible. This situation may occur, for example, because a ship is leaving the theater of operations, or is heading into an in-theater port for extensive repair work. Such a platform will be designated in the formulation as an "expend" platform, and, with all other factors being equal, the model will assign to it the maximum possible tasking. The next priority in missile spreading is to level assets across the non-expend platforms. The model will attempt to leave all non-expend firing platforms with the same number of missiles; this prevents depleting the magazine of any non-expend firing platform. Finally, as a tactical concern, it may be desired to spread primary and/or back-up missiles across as many firing platforms within a launch area as possible, or, conversely, to concentrate assigned tasking on as few platforms as possible. This accounts for the trade-off between minimizing the potential damage of a single point failure, and minimizing the number of platforms that are required for TLAM tasking. The option to spread task parts individually (i.e., primary missiles but not back-ups, or vice versa) is allowed; this gives the TSC flexibility in cases where a primary to back-up ratio other than 1:1 is desired. Such a case may arise, for example, if a number of back-ups are tasked that have no associated primary missile. This form of tasking is used to enable the rapid execution of strikes that must be planned for, but for which the TSC has not yet received launch authority.

C. MODEL PRESENTATION

The data in the formulation is derived from the data and tactical options enumerated in the description of the overall TSC predesignation system. The primary decision variables for the formulation are binary variables, indexed over firing platforms, half-modules, cells, tasks, and task parts (primary, ready-spare, or back-up). These variables represent the selection, or non-selection, of a missile on board firing platform f, in half-module h and cell c, to meet task t, part p.

The objective function is presented in order of solution priorities. Priorities are assigned as follows, from highest to lowest:

a. Meet all assigned tasking through the use of a penalty assessed for any unmet task parts. Primary missiles are considered the most important to select, followed by back-up missiles, and then ready-spares;
b. Minimize employment penalties, which quantify the opportunity cost of using a platform for TLAM operations instead of other tasking;
c. Maximize missiles allocated from designated expend platforms;
d. Level the number of missiles remaining on non-expend platforms;
e. Spread primary missiles among as many firing platforms as possible, in launch areas so designated;
f. Spread back-up missiles among as many firing platforms as possible, in launch areas so designated;
g. Minimize the M^3 list position of all selected missiles, ensuring that, all other priorities being satisfied, the least capable missile is allocated for each mission; and, finally,
h. Maximize residual salvo capability.

Although the ordering of the priorities presented in this thesis was developed jointly with NSWCDD [Fennemore, 1998c], and was informally reviewed by members of Tactical Training Group Atlantic [Williams, 1998], it does not reflect any official guidance.

The constraints assign values to the accounting variables used in the objective function and ensure that a solution is operationally feasible. General constraints are imposed to govern the way in which missiles may be chosen for tasking, and ships used to handle tasking requirements. Noteworthy operational requirements are as follows: (a) Only one cell per half-module is allowed to be selected; this is due to a design characteristic of the Vertical Launch System (VLS); (b) Ready-spare missiles for a task can only be assigned to the firing platform to which the primary missile for that task has been assigned; and (c) back-up missiles can only be assigned to a firing platform other than the one to which the primary missile for that task has been assigned. The general format of several constraint sets is derived from Kuykendall [1998].

A more specific description of each term of the objective function and each constraint follows the formulation.

FORMULATION

Indices:

w	weapon type loaded in cell, including block and variant (e.g., CIII, DII).
f	firing platform (e.g., DDG-57, CG-73, DD-997).
h	half-module, number is dependent on type of ship (e.g., h1-h24 for DDG-57).
c	cell, each half-module contains four cells (c1-c4) or (c5-c8).
t	task number (e.g., t1, t2, …); each task number corresponds to a mission ID number. Mission ID numbers may have multiple associated tasks.
p	task part, each task may consist of a primary part, a ready-spare part, and a back-up part indexed as follows: 1 = primary, 2 = ready-spare, 3 = back-up.
a	launch area, i.e., geographic location with multiple FPPWP in close proximity (e.g., N. RED SEA, EAST MED).
i	instance of a set of conflicting tasks (e.g., {t1, t2, t3})

Set:

con_i	set of tasks that are to be executed within a period of time such that all tasks in the set conflict, i.e., only one task per set can be assigned to any one half-module, where con_i denotes a set of conflicting tasks.

Binary Data:

inArea_{ta}	equals 1 if task t is in launch area a, 0 otherwise
geoFe_{ft}	equals 1 if platform f can attain firing position for task t, 0 otherwise
numReq_{tp}	equals 1 if task t, part p requires a missile, 0 otherwise
expend_f	equals 1 if it is desired to expend as many missiles as possible from platform f, 0 otherwise

priSpread_a	equals 1 if it is desired to spread primary missiles across as many platforms in launch area *a* as possible, 0 otherwise
buSpread_a	equals 1 if it is desired to spread back-up missiles across as many platforms in launch area *a* as possible, 0 otherwise
load_{wfhc}	equals 1 if a weapon of type *w* is loaded in location (*f,h,c*), 0 otherwise
empPen_f	equals 1 if it is desired to have firing platform *f* continue its current mission instead of being used for TLAM tasking, 0 otherwise

Weighting Data:

mCubePri_p	weight of M^3 list position for task part *p*

Numerical Data:

levelPlat	total number of firing platforms in the operational theater minus the number of "expend platforms", i.e., the group of platforms across which it is desired to level the number of missiles after a strike
mCubePos_{wt}	M^3 list position of weapon type *w* for task *t* if the M^3 list for task *t* contains weapon type *w*, 0 otherwise
value_w	relative value of weapon type *w* (e.g., CIII are more valuable than CII, which are more valuable than DIII)

Derived Data:

avail_{fhctp}	equals 1 if firing platform *f*, half-module *h*, cell *c* contains a TLAM available for assignment to meet part *p* of task *t*, 0 otherwise
avail_{ftp}	$\{(h,c) \mid \mathrm{avail}_{fhctp} = 1\}$
geoFe	$\{(f,t) \mid \mathrm{geoFe}_{ft} = 1\}$
$\mathrm{mCubePos}_t$	$\{w \mid \mathrm{mCubePos}_{wt} = 1\}$
load_w	$\{(f,h,c) \mid \mathrm{load}_{wfhc} = 1\}$
inArea_t	$\{a \mid \mathrm{inArea}_{ta} = 1\}$
inArea_a	$\{t \mid \mathrm{inArea}_{ta} = 1\}$

Binary Variables:

X_{fhctp}	equals 1 if weapon in location (*f,h,c*) is selected for task *t*, part *p*, 0 otherwise
UNABLE_{tp}	equals 1 if missile cannot be allocated for task *t*, part *p*, 0 otherwise
$\mathrm{SHIPREQ}_{fa}$	equals 1 if platform *f* is required to be in area *a* for firing, 0 otherwise
SALVO_{wfh}	equals 1 if one or more weapons of type *w* remain on platform *f* in half-module *h* after firing all primary missiles, 0 otherwise
$\mathrm{NUMPLATPR}_f$	equals 1 if platform *f* is assigned one or more primary task parts, 0 otherwise
$\mathrm{NUMPLATBU}_f$	equals 1 if platform *f* is assigned one or more back-up task parts, 0 otherwise

Counting Variables:

NUMEXPEND	number of missiles selected from platforms designated to receive as much tasking as possible
NUMREMAIN_f	number of missiles on platform *f* that are not selected for a primary task part from non-expend platforms
AVGREMAIN	mean number of missiles of all types on board “non-expend” firing platforms in theater, after all missiles have been selected for the given strike
DIFFMEAN_f	difference between the residual number of missiles on platform *f* and AVGREMAIN
MCUBESUM_p	sum of M^3 list positions for missiles that are selected for part *p*

Formulation:

MINIMIZE

1a) $\sum_{tp} \mathrm{UNABLE}_{tp}$

1b) $+ \sum_{fa} (\mathrm{empPen}_{f} \ * \ \mathrm{SHIPREQ}_{fa})$

1c) $-$ NUMEXPEND

1d) $+ \sum_{f} \mathrm{DIFFMEAN}_{f}$

1e) $- \sum_{f} \mathrm{NUMPLATPR}_{f}$

1f) $- \sum_{f} \mathrm{NUMPLATBU}_{f}$

1g) $+ \sum_{p} \mathrm{mCubePri}_{p} \ * \ \mathrm{MCUBESUM}_{p}$

1h) $- \sum_{wfh} (\mathrm{value}_{w} \ * \ \mathrm{SALVO}_{wfh})$

Subject to:

2) $\sum_{f \mid \mathrm{geoFe}_{ft}=1} \ \sum_{hc \mid \mathrm{avail}_{fhctp}=1} X_{fhctp} \le \mathrm{numberReq}_{tp} \qquad \forall \ (t,p) \mid \mathrm{numReq}_{tp} = 1$

3) $X_{fhct"1"} \le \mathrm{SHIPREQ}_{fa} \qquad \forall \, (f,t) \mid \mathrm{geoFe} = 1$

$a \mid \mathrm{inArea}_{t} = 1$

$(h,c) \mid \mathrm{avail}_{ft"1"} = 1$

$$4)\ X_{\text{fhct"3"}} \le \text{SHIPREQ}_{\text{fa}} \qquad \begin{array}{l} \forall\, (f,t) \mid \text{geoFe} = 1 \\ a \mid \text{inArea}_t = 1 \\ (h,c) \mid \text{avail}_{\text{ft"1"}} = 1 \end{array}$$

$$5)\ \sum_{a} \text{SHIPREQ}_{\text{fa}} \le 1 \qquad \forall\, f$$

$$6)\ \sum_{f \mid \text{expend}_f = 1} \ \sum_{hct \mid \text{avail}_{\text{fhct"1"}} = 1} X_{\text{fhct"1"}} = \text{NUMEXPEND}$$

$$7)\ \sum_{whc} \text{load}_{\text{wfhc}} - \sum_{hct \mid \text{avail}_{\text{fhct"1"}} = 1} X_{\text{fhct"1"}} = \text{NUMREMAIN}_f \qquad \forall\, f \mid \text{expend}_f = 0$$

$$8)\ \frac{1}{\text{levelPlat}} * \sum_{f} \text{NUMREMAIN}_f = \text{AVGREMAIN}$$

$$9)\ (\text{NUMREMAIN}_f - \text{AVGREMAIN}) * (1 - \text{expend}_f) \ge -\text{DIFFMEAN}_f \qquad \forall\, f \mid \text{expend}_f = 0$$

$$10)\ (\text{NUMREMAIN}_f - \text{AVGREMAIN}) * (1 - \text{expend}_f) \le \text{DIFFMEAN}_f \qquad \forall\, f \mid \text{expend}_f = 0$$

$$11)\ \sum_{a \mid \text{priSpread}_a = 1} \ \sum_{t \mid \text{inArea}_a = 1} \ \sum_{hc \mid \text{avail}_{\text{fhct"1"}} = 1} X_{\text{fhct"1"}} \ge \text{NUMPLATPR}_f \qquad \forall\, f$$

$$12)\ \text{NUMPLATPR}_f \le 1 \qquad \forall\, f$$

$$13)\ \sum_{a \mid \text{buSpread}_a = 1} \ \sum_{t \mid \text{inArea}_a = 1} \ \sum_{hc \mid \text{avail}_{\text{fhct"3"}} = 1} X_{\text{fhct"3"}} \ge \text{NUMPLATBU}_f \qquad \forall\, f$$

$$14)\ \text{NUMPLATBU}_f \le 1 \qquad \forall\, f$$

$$15)\ \sum_{w \mid \text{mCubePos}_t = 1} \ \sum_{fhc \mid \text{load}_w = 1} \ \sum_{t \mid \text{avail}_{\text{fhctp}} = 1} X_{\text{fhctp}} = \text{MCUBESUM}_p \quad \forall\, p$$

$$16)\ \sum_{c \mid \text{avail}_{\text{fhct"1"}} = 1} \text{load}_{\text{wfhc}} - \sum_{ct \mid \text{avail}_{\text{fhct"1"}} = 1} X_{\text{fhctp}} \ge \text{SALVO}_{\text{wfh}} \qquad \forall\, (w,f,h) \mid \sum_{c} \text{load}_{\text{wfhc}} \ge 1$$

$$17)\quad \sum_{w} \text{SALVO}_{wfh} \le 1 \qquad \forall\,(f,h) \mid \sum_{cw} \text{load}_{wfhc} \ge 1$$

$$18)\quad \sum_{hc \mid \text{avail}_{\text{fhct"1"}}=1} X_{\text{fhct"1"}} \ge \sum_{hc \mid \text{avail}_{\text{fhct"2"}}=1} X_{\text{fhct"2"}} \qquad \forall\,(f,t) \mid \text{geoFe}_{ft} = 1$$

$$19)\quad \sum_{c \mid \text{avail}_{\text{fhct"1"}}=1} X_{\text{fhct"1"}} + \sum_{c \mid \text{avail}_{\text{fhct"2"}}=1} X_{\text{fhct"2"}} \le 1 \qquad \forall\,(f,h,t) \mid (\sum_{cw} \text{load}_{wfhc} \ge 1 \cup$$
$$t \mid (\text{numReq}_{t\text{"1"}} + \text{numReq}_{t\text{"2"}}) = 2)$$

$$20)\quad \sum_{hc \mid \text{avail}_{\text{fhct"1"}}=1} X_{\text{fhct"1"}} + \sum_{hc \mid \text{avail}_{\text{fhct"3"}}=1} X_{\text{fhct"3"}} \le 1 \qquad \forall\,(f,t) \mid \text{geoFe}_{ft} = 1$$

$$21)\quad \sum_{tp \mid \text{avail}_{fhctp}=1} X_{fhctp} \le 1 \qquad \forall\,(f,h,c) \mid \sum_{w} \text{load}_{wfhc} \ge 1$$

$$22)\quad \sum_{fhc \mid \text{avail}_{fhctp}=1} X_{fhctp} \le 1 \qquad \forall\,(t,p) \mid \text{numReq}_{tp} = 1$$

$$23)\quad \sum_{cp \mid \text{avail}_{fhctp}=1} \sum_{t \in \text{con}_i} X_{fhctp} \le 1 \qquad \forall\,(f,h) \mid \sum_{cw} \text{load}_{wfhc} = 1, \forall\, i$$

NOTES

Objective function explanation:

1a) Expresses the penalty for failure to meet tasking requirements,

1b) plus the sum of employment penalties of platforms being used for Tomahawk tasking instead of continuing the mission they were performing,

1c) minus the benefit for the use of missiles from designated "expend" platforms,

1d) plus a penalty for differences between the number of missiles remaining on non-expend platforms after tasking, and the average number of missiles remaining on non-expend platforms after tasking,

1e) minus the benefit of spreading primary missiles across as many platforms as possible, where it is desired to do so,

1f) minus the benefit of spreading back-up missiles across as many platforms as possible, where it is desired to do so,

1g) plus a penalty that increases with the M^3 list position of the selected missile, and

1h) minus the total residual salvo capability for each weapon type.

Constraint explanations:

2) The sum of missiles selected and the number of missiles that cannot be selected is equal to the number of missiles required for each task and part. The variable X_{fhctp} is only "counted" for missile selection if, for a specified launch area, the task is in the area, and the firing platform can attain launch position in the area.

3) A firing platform *f* is required to be in area *a* if a missile on that firing platform is selected for a primary task in area *a*.

4) A firing platform *f* is required to be in area *a* if a missile on that firing platform is selected for a back-up task in area *a*.

5) A platform *f* can launch from no more than one area.

6) The variable $NUMEXPEND_f$ equals the sum of all missiles selected from expend platforms.

7) The variable $NUMREMAIN_f$ equals the initial load-out of missiles minus the sum of all primary missiles fired, for each platform that is not designated expend. This assumes that only primary missiles are fired, and that all primary missiles are fired.

8) The average number of missiles remaining on non-expend platforms across all platforms in theater is equal to the sum of NUMREMAIN divided by the total number of non-expend platforms in theater.

9-10) The variable $DIFFMEAN_f$ is set for each "nonexpend" firing platform by comparing the residual number of missiles to the average number of remaining missiles.

11-12) The variable $NUMPLATPR_f$ equals one for each firing platform on which primary missiles are selected, for launch areas that require primary spreading.

13-14) The variable $NUMPLATBU_f$ equals one for each firing platform on which back-up missiles are selected, for launch areas that require back-up spreading.

15) The variable MCUBESUM equals the sum of M^3 list positions for all missiles that are selected.

16) The variable $SALVO_{wfh}$ is restricted to equal zero if all missiles in a half-module have been expended, for each firing platform and weapon type.

17) The salvo size for a weapon type for each firing platform and half-module is either 0 or 1. This is due to half-module power-up constraints – a surface ship may only apply power to one cell per half-module. The formulation will only credit the follow-on salvo capability with at most one weapon of each type per half-module.

18) Ready-spare missiles for a task must be assigned to the same firing platform as the primary missile for that task, for each firing platform and task.

19) Ready-spare missiles for a task cannot be assigned to the same half-module as the primary missile for that task, for each firing platform and task.

20) Back-up missiles for a task must be assigned to a firing platform other than that assigned the primary missile for that task, for each firing platform and task.

21) Only one missile may be selected per half-module on a given firing platform.

22) Only one missile may be selected per task part.

23) Only one missile per set of conflicting tasks may be selected per half-module for each firing platform.

D. IMPLEMENTATION

The TSC predesignation model is implemented in GAMS version 2.50 [GAMS, 1999] using the CPLEX solver version 6.5 [ILOG, 1999], and run on a 333 MHz PC with a Pentium II processor. Formulation alternatives were tested to improve solution time and the quality of the solution (e.g., within 5% - 10% of a provably optimal solution).

1. Weighted Objective Function Implementation

Initially, the model was implemented as a monolithic problem, with a single objective function containing all terms. Each term (e.g., using missiles from expend platforms, or maximizing residual salvo capability) represents a distinct priority to be considered in allocating weapons and firing platforms. Numerical multipliers, or weights, preserve the ordering of the priorities of the solution. For example, the highest priority of the formulation is to meet all assigned tasking. The weight applied to this objective function term is of sufficient magnitude to ensure that no combination of allocations achieved with lesser priorities can outweigh an allocation with the least amount of unmet tasking possible.

The most serious problem with using objective function weights to enforce prioritized solutions is that of inaccuracy resulting from non-zero integrality gaps (i.e., the ratio between the best integer solution found when the algorithm terminates and the bound on the best possible solution). An example of the process of determining weights is presented to illustrate how this inaccuracy occurs.

In the TSC predesignation formulation, the lowest priority is residual salvo capability. Each missile of a specific type is assigned a relative value. Because this is the lowest priority, the numbers can be small, such as assigning a CII missile a value of one, and a CIII missile a value of two. This allows for a sufficient number of less capable missiles to be considered more valuable than a small number of more capable missiles; this is an intentional effect of the formulation, and places a premium on overall striking power rather than on the ability to accomplish a specific set of missions. With some assumptions about the number of firing platforms under consideration, and the expected number of TLAM's available, the maximum possible value for the SALVO variable can be determined. Using the assumptions specific to the scenarios run in this thesis, discussed in detail in Chapter III, the maximum possible value of SALVO is:

$$\sum_{w} (\text{value}_w)(\text{number}_w)$$

where *w* is the type of missile, value_w is the value of a type *w* missile, and number_w is the number of missiles of type *w*

$$= (2)(104) + (1)(78)$$

$$=280.$$

The next priority is the sum of all M^3 list positions for the selected missiles. This priority is further separated into task parts, so that primary missile M^3 list positions are considered more important than back-ups, and back-ups are more important than ready-spares. Using the maximum possible value for residual salvo capability of 280, a weight which is slightly larger, 300, is applied to each M^3 list position of missiles selected for ready-spare missions. This weight ensures that residual salvo considerations will not affect the selection of missiles to minimize M^3 list position. Assumptions about the maximum possible tasking, and the M^3 list for tasks gives a worst case scenario of 64 ready-spare missiles being selected, all with M^3 list position two. The maximum possible total of the sum of M^3 positions for ready-spares is then:

> where M^3 Weight is the weight for M^3 list positions of missiles selected for ready-spares, j is an index of M^3 list positions (e.g., one or two), and
>
> $$M^3 \text{ Weight} \sum_{j} (j * \text{number}_j)$$
>
> number_j is the number of missiles selected that have M^3 list position j
>
> $= (300)(2)(64)$
>
> $= 38{,}400.$

This number then influences the selection of the weight on the objective function term for M^3 list positions for back-up missiles. This weight must account for the sum of all previous worst case scenarios, and so must be larger than 38,400 + 280, or 38,680. The remaining weights are calculated in a similar manner. For the scenarios used in the thesis, this results in the highest priority weight, i.e., the penalty for unmet primary tasking, of $8.32\text{x}10^{19}$.

This becomes problematic when the solver is trying to differentiate between solutions. In order to allow the solver to find a solution in an amount of time which represents the desired balance between the required quality of the solution and the time needed to return an optimal solution within the limits of machine precision, a relative integrality gap is set; this gap permits the solver to return a solution which, while not optimal, is within the specified tolerance of being optimal. A model that has a wide range of objective function weights is in danger of having the lower weights largely ignored by the solver; with a relative gap greater than $1.0\text{x}10^{-15}\%$, an objective function value of $8.32\text{x}10^{19}$ is indistinguishable from a value of $8.32\text{x}10^{19} + 280$. Using the formulation presented in this thesis, a small scenario assigning 29 missiles to four firing platforms with a combined salvo capability of 104 TLAM's, an allowed relative integrality gap of 5% led to a solution with an absolute integrality gap of 5.5×10^{16}. The effect of this gap was seen in the variables for the lower priority objective function terms. The model selected missiles that resulted in a sub-optimal solution for considerations such as the sum of M^3 positions and residual salvo capability.

An additional difficulty with the method of weighted objective functions is one of numerical stability. Every computer has limitations on the precision it maintains in memory while performing arithmetic operations. The computation necessary to execute the branch and bound algorithm on problem instances with numbers (i.e., weights) differentiated by large orders of magnitude arguably approaches the limits of precision of the computer used.

2. Hierarchical Restriction Implementation

Based on the poor performance of the weighting, we have abandoned weighting in favor of multiple objective function hierarchical restriction. The monolithic objective function is separated into eight sub-problems, each containing a single-term objective function; these sub-problems are solved in decreasing order of objective function priority. Optimal values for the objective function term variables obtained in one sub-problem are fixed, and provided to the subsequent sub-problem as input. By allowing the primary variable, individual missile selection, to change while maintaining the values of objective function term variables set in previous sub-problems, it is possible to solve for each priority, ensuring that no lesser consideration can impact a more important concern. For example, the highest priority of the model is to meet all assigned tasking. Using the hierarchical restriction formulation, we solve the first sub-problem to minimize failure to meet tasking for each task and part. Once the optimal solution is found, the values for these variables are set. These values are passed to the next sub-problem as data. This model, which minimizes any employment penalties, is then solved, with a constraint that unmet tasking is fixed according to the solution from the first sub-problem. Employment penalties are then fixed to the values returned by the second sub-problem. The next sub-problem, which maximizes the allocation of missiles to any expend platforms, receives the unmet tasking and employment penalty data, and is constrained to maintain these values. The process is continued until all sub-problems have been optimized.

Hierarchical restriction also simplifies the problem by allowing us to consider a restricted subset of the constraints. Specifically, with a single objective function term, the only constraints that need to be considered are those that relate to the objective function variables under consideration, those that fix variables whose values have been set in a previous sub-problem, and constraints that enforce restrictions on missile selection. This provides a significant reduction in the number of rows and columns generated for each model. The hierarchical restriction sequence is summarized in Table 1; the objective function term and constraint numbers are consistent with those given in the formulation.

HIERARCHICAL RESTRICTION SEQUENCE			
Sub-problem:	Objective Function:	Subject to constraints:	Variable to be fixed:
1	Minimize 1a	2, 3, 4, 5, 18, 19, 20, 21, 22, 23	$UNABLE_{tp}$
2	Minimize 1b	2, 3, 4, 5, 18, 19, 20, 21, 22, 23	$SHIPREQ_{fa}$
3	Maximize 1c	2, 3, 4, 6, 18, 19, 20, 21, 22, 23	NUMEXPEND
4	Minimize 1d	2, 3, 4, 6, 7, 8, 9, 10, 18, 19, 20, 21, 22, 23	$DIFFMEAN_f$
5	Maximize 1e	2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 18, 19, 20, 21, 22, 23	$NUMPLATPR_f$
6	Maximize 1f	2, 3, 4, 6, 7, 8, 9, 10, 11, 13, 14, 18, 19, 20, 21, 22, 23	$NUMPLATBU_f$
7	Minimize 1g	2, 3, 4, 6, 7, 8, 9, 10, 11, 13, 15, 18, 19, 20, 21, 22, 23	$MCUBESUM_p$
8	Maximize 1h	2, 3, 4, 6, 7, 8, 9, 10, 11, 13, 15, 16, 17, 18, 19, 20, 21, 22, 23	$SALVO_{wfh}$

Table 1: Hierarchical Restriction Sequence. Sub-problems are numbered in decreasing order of priority. The objective function and constraint numbers refer to the formulation. Each sub-problem is solved and the variables that appear in the objective function are fixed. Constraints are added to each sub-problem to ensure that allocations are operationally feasible.

Hierarchical restriction allows us to more accurately capture priorities at all levels given hardware and software limitations. However, our implementation of the method does not necessarily provide optimal solutions. By fixing the values of the relevant variables after the solution of each sub-problem, we are precluding solutions in subsequent sub-problems which may result from an alternate optimal solution in a previously considered sub-problem. This difficulty can be avoided by constraining the objective function value derived from solving a given sub-problem to be (almost) as good as its value when solving a subsequent sub-problem (e.g., [Steuer, 1986], pp. 292-296). For example, if solving sub-problem 1 yields three primary task parts that cannot be met, with our method, the value of the unmet tasking variable is set equal to one for the three task parts returned in the optimal solution. Alternately, we could add a constraint ensuring that among all tasks, three primary missiles cannot be allocated. The drawback to the latter approach manifests itself in resulting sub-problems with a greater number of variables and constraints than with the sub-optimal method we used. In fact, the size of our problem rendered the optimal method unusable. Even with our implementation, as the size of the scenario increases, there is a marked increase in solution time. This is presented in Chapter III, but the solution times for even moderately sized scenarios are poor enough to mandate further implementation refinements. We next present a heuristic variation of our hierarchical restriction procedure to improve solution times, albeit at the expense of some solution quality.

3. Heuristic Structure

The hierarchical restriction implementation results in exceedingly long solution times for certain sub-problems in various problem scenarios (e.g., on the order of 2,000 seconds for a single sub-problem in some cases). Based on observations of the behavior of the model, we implement a heuristic procedure in conjunction with the hierarchical restriction methodology. The logic for the heuristic is based on four ideas:

First, for several of the sub-problems, it may be possible to decrease solution time by solving a sub-problem multiple times, each time using a restricted set of the data. Specifically, solving successively for individual task parts (i.e., primary, ready-spare, and back-up) rather than solving for all task parts simultaneously reduces solution times for some of the sub-problems by reducing the sub-problem size. Solving for separate task parts is used for sub-problems one and seven, which address unmet tasking and the sum of M^3 list positions respectively. For sub-problem one, if all required allocations can be made using this method, then the unmet tasking variables are fixed to zero, the allocation variables are released, and the model continues to the next sub-problem. For sub-problem seven, if a feasible solution is found for all tasking with the sequentially fixed allocation, then the M^3 list position variables are fixed, the allocations released, and the model proceeds to the next sub-problem. In both cases, if the required tasking cannot be met with this method, the sub-problem is re-solved for all task parts together, i.e., with the full-scale hierarchical restriction implementation.

Second, obtaining a solution using only a subset of data in one sub-problem may enable us to omit a subsequent sub-problem entirely. This method is used in instances of sub-problem three in which there are firing platforms that are assigned an employment penalty. If all tasking can be allocated in the first sub-problem without using any of these firing platforms, then the value for employment penalties is set to zero, and the sub-problem for employment penalties is skipped.

Third, we observe that for some sub-problems, almost all of the solution time is spent trying to find a feasible solution. If this initial feasible solution is not within the specified optimality tolerance, the solver finds one that meets the tolerance within a matter of seconds. Relatively few constraints are added to each successive sub-problem. Additionally, all operational constraints are met in each sub-problem (thus ensuring that any feasible allocation determined from a previous sub-problem is also feasible for a subsequent sub-problem). Thus, the allocation from the previous sub-problem may provide a relatively good solution to the subsequent sub-problem.

This suggests the following method for reducing solution time. Rather than solving a sub-problem requiring a long solution time and whose objective function value may not differ substantially if it is evaluated with the allocation variables fixed from the previous sub-problem, we proceed as follows: Fix the allocation variables to the values from the preceding sub-problem. Next, compute the values for the corresponding objective function variables. The values for allocation variables are then released, and the next sub-problem is solved. This results in a significant decrease in computation time, because the solver is performing a simple arithmetic computation rather than a potentially large mixed integer optimization.

The method of fixing allocations after a preceding sub-problem and computing the objective function value rather than optimizing is used for sub-problems five and six, corresponding to the spreading of primary and back-up missiles, and sub-problem eight, which maximizes residual salvo capability.

Finally, a simple if-then check may be used to determine the need to solve sub-problems. If a sub-problem is associated with an operator selection, and the operator does not desire to perform the action in question, then the sub-problem does not need to be run. This method is employed with sub-problem three, which maximizes allocations to expend platforms. If no expend platforms are designated, the sub-problem is omitted. This method could also be applied to sub-problems five and six; however, the time saved by omitting the sub-problem rather than computing the objective function value for a fixed allocation is negligible.

The structure of the heuristic implementation is presented in Figure 7. The conditional and procedural statements reflect the optimization of a particular sub-problem, whether the missile allocation is treated as fixed or variable, and the necessity of solving a particular sub-problem. The task blocks for solving sub-problems one and seven are simplified; they do not represent the process of solving for each task part separately. Figures 8 and 9 give detailed procedures of the heuristic method for sub-problems one and seven, respectively.

There are several caveats associated with the use of this heuristic. Efficiency is highly problem-specific. In particular, the use of restricted data sets does not ensure a faster solution time; in fact, it is theoretically possible to construct cases for which the solution time using the heuristic is longer than that of the hierarchical restriction implementation, if the restricted data sets do not render a feasible solution. For example, the solver may spend time determining that tasking cannot be met without using a platform with an employment penalty, and the entire sub-problem must be re-run considering the additional firing platforms.

Additionally, the solution obtained using the heuristic will be no better than the hierarchical restriction solution, and, in some cases, may be worse. This is because allocations are fixed in a greedy and myopic manner. For example, in solving for M^3 list position by task part, it is possible that primary allocations will be made that will adversely impact the choice of ready-spare missiles with respect to M^3 list position, and the combination of fixed primary and ready-spare allocations will similarly adversely affect the selection of back-up missiles. Further, the decision to temporarily fix all

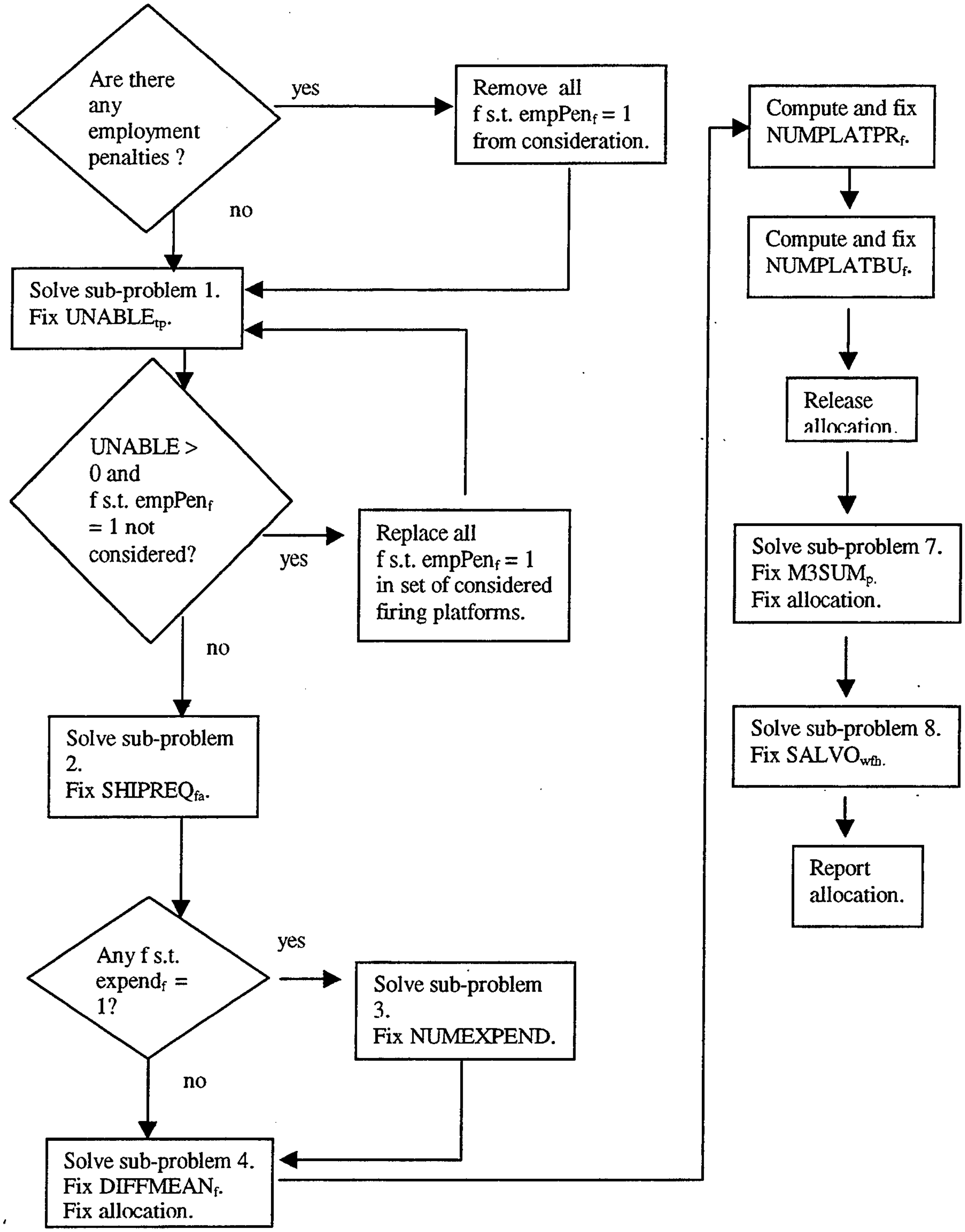


Figure 7: Heuristic Solution Method. Notation is consistent with the formulation. See Figures 8 and 9 for expanded views of sub-problems 1 and 7.

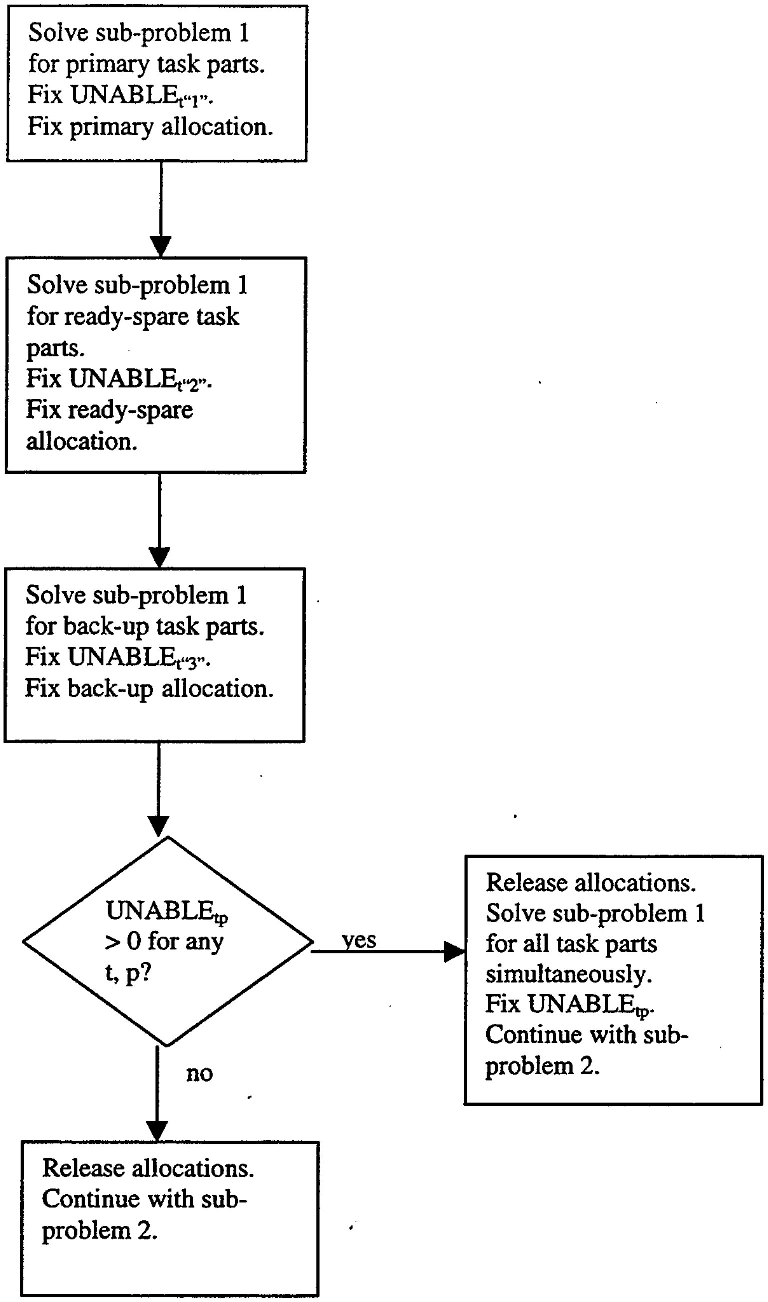


Figure 8: Expanded Heuristic for Sub-Problem 1 to Minimize Unmet Tasking. See Figure 7 for complete heuristic flow.

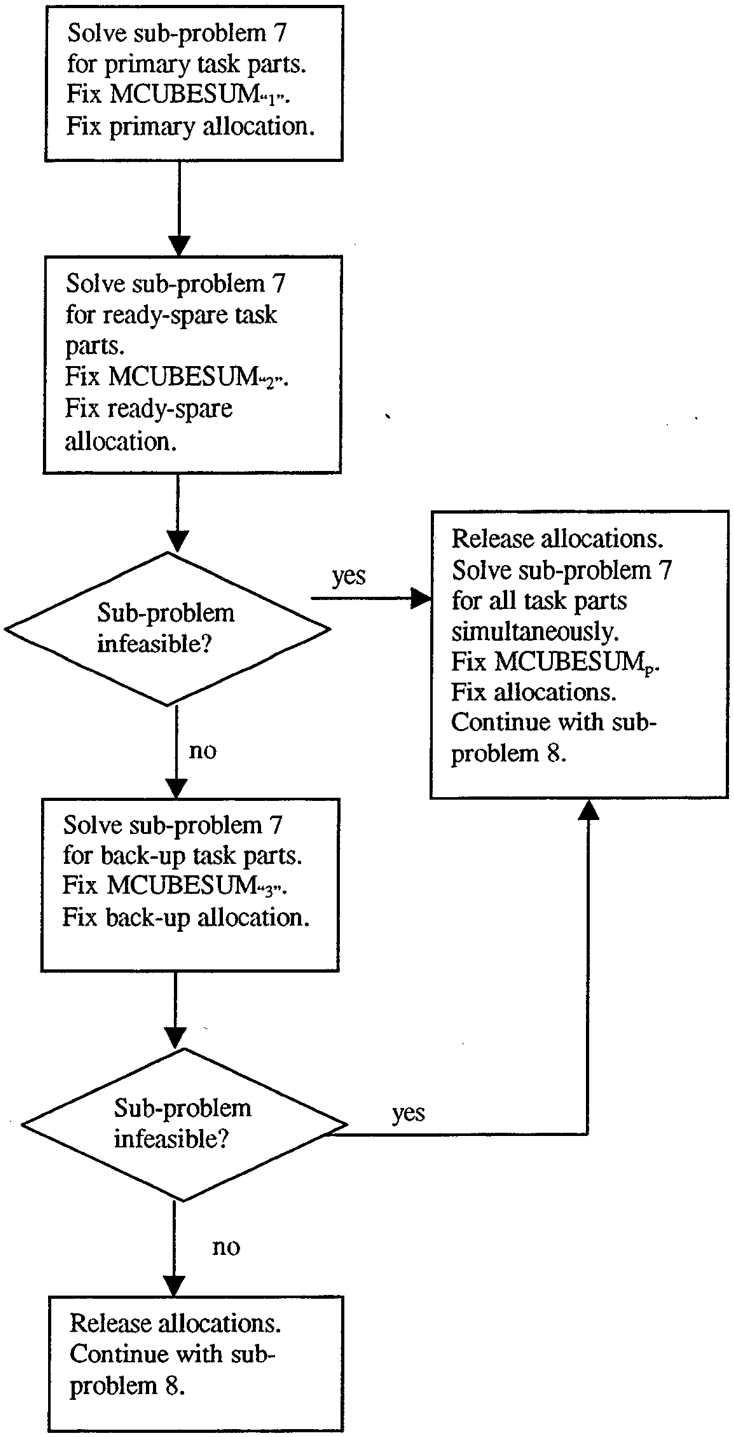


Figure 9: Expanded Heuristic for Sub-Problem 7 to Minimize M^3 List Position. See Figure 7 for complete heuristic flow.

allocations before calculating the number of platforms primary and back-up missiles are spread across, and residual salvo capability is one that implicitly accepts a sub-optimal solution in favor of greatly reduced solution times.

Results for scenarios run using the heuristic are given in Chapter III. Solution quality appears to be acceptable for all cases tested. Note, however, that the heuristic has been designed expressly for these particular scenarios, and additional procedures (e.g., subroutines and condition checks) would be required before it could be applied to scenarios differing substantially from those presented here.

THIS PAGE INTENTIONALLY LEFT BLANK

III. RESULTS

A. SCENARIO DESCRIPTIONS

Scenarios for demonstrating the TSC Predesignation model are based on data provided by NSWCDD [Fennemore, 1999b]. There are six “base” scenarios, which are designated numerically (i.e., scenarios 1 – 6). The base scenarios show the effectiveness of the model as the amount of tasking increases. Five additional scenarios, denoted alpha-numerically (e.g., 1A), are presented to exercise all the options that the formulation allows. The additional, or modified, scenarios have the same amount of tasking as the associated base scenario, but demonstrate allocations resulting from different operator input. For example, Scenario 1 requires 13 primary missiles, 8 ready-spare missiles, and 8 back-up missiles. Scenario 1A requires the same amount of tasking, but also designates an expend platform, which is not required in Scenario 1.

Scenario 6 introduces the concept of “fixed tasking”. This assumes that the tasking from Scenario 5 has already been assigned to firing platforms and missiles have been selected to meet this tasking. Additional tasking is then allocated, with the restriction that the previous allocation must remain unchanged. This capability is demonstrated at the request of NSWCDD, and represents a first step towards a robust predesignation system that is capable of performing allocations under varying operational conditions.

The scenarios are summarized in Table 2.

SCENARIO	NUMBER OF FIRING PLATFORMS			TASKS ASSIGNED			f s.t. expend$_f$ = 1	MISSILE SPREAD		NUMBER OF LAUNCH AREAS	f s.t. empPen$_f$ = 1
	CG	DDG	DD	PRI	RS	BU		PRI	BU		
1	2	1	1	13	8	8	0	Y	N	1	1 CG
1A	2	1	1	13	8	8	1	N	N	1	1 CG
1B	2	1	1	13	8	8	0	N	Y	1	NONE
2	2	1	1	26	16	16	0	Y	N	1	1 CG
2A	2	1	1	26	16	16	0	Y	N	2	1 CG
3	2	1	1	52	32	32	0	Y	N	1	1 CG
4	2	1	1	65	40	40	0	Y	N	1	1 CG
5	3	2	2	78	48	48	0	Y	N	1	1 CG
5A	3	2	2	78	48	48	3	Y	N	1	1 CG
5B	3	2	2	78	48	48	0	Y	N	1	2 CG, 1 DD
6	3	2	2	104	64	64	3	Y	N	1	1 CG

Table 2: Scenario Descriptions.

Notes: 1. CG is used to denote a TICONDEROGA class cruiser. DDG indicates an ARLEIGH BURKE class destroyer, and DD refers to a SPRUANCE class destroyer.

2. PRI, RS, and BU represent primary, ready-spare, and back-up task parts, respectively.

3. Data sets expend$_f$ and empPen$_f$ refer to the formulation.

4. Scenario 1A demonstrates the use of an expend platform.

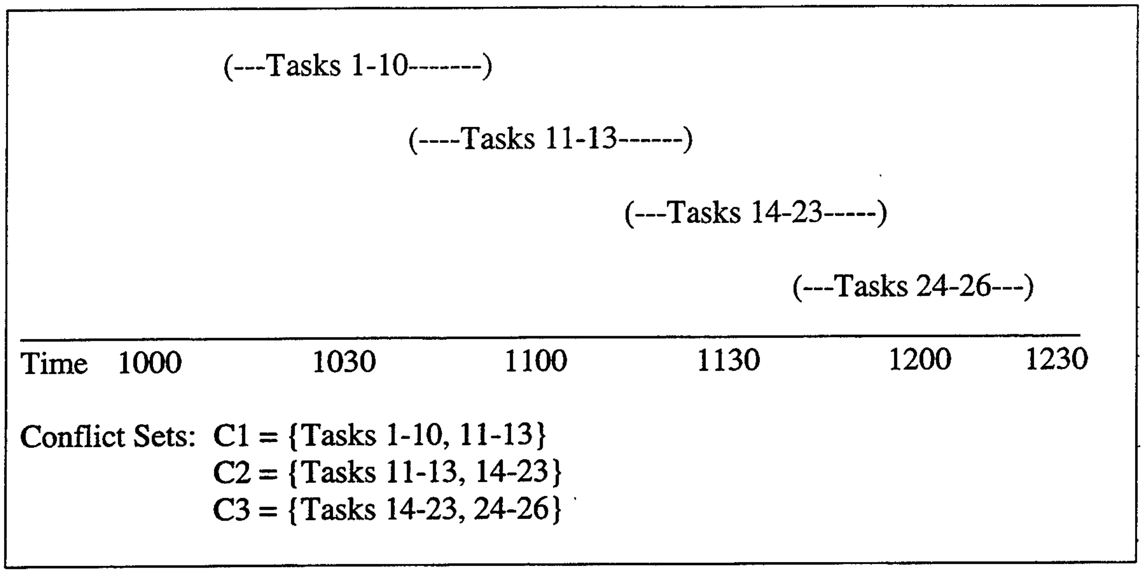
5. Scenario 1B illustrates the spreading of back-up missiles among as many firing platforms as possible.

6. Scenario 2A has two launch areas. One CG and one DD can only perform tasking in one of these areas; one CG and one DDG may only be assigned tasking in the other area.

7. Scenarios 5A and 5B show the effects of expend platforms and employment penalties, respectively, with larger sets of tasking.

8. In Scenarios 5, 5A, 5B, and 6, thirteen of the tasks may only be allocated to three specific firing platforms. This illustrates the possibility of firing platforms not having mission data or the correct missile type for certain missions.

For each scenario, tasks are given in groups of thirteen. Each task group requires thirteen primary, eight ready-spare, and eight back-up missiles. Using an assumed preparation time for a TLAM of approximately 45 minutes, the launch times for tasks are assigned so that all thirteen tasks in a group "conflict" with every other task in that group. This means that, because the missiles are being prepared for launch at the same time, only one missile per conflict set may be assigned to a half-module. Additionally, the last three tasks in each group conflict with the first ten tasks of the following group. For example, Scenario 2 assigns 26 tasks, in two groups of thirteen. Task numbers one through ten have a notional launch time of 1100 on the day of the scenario; task numbers eleven through thirteen have a launch time of 1130. Tasks fourteen through twenty-three are to be launched at 1200, and the final three tasks each have a launch time of 1230. With the assumed preparation time of 45 minutes prior to launch, this scenario generates the following conflict sets: Tasks 1-13, Tasks 11-23, and Tasks 14-26. Figure 10 gives a graphical representation of how conflict sets are formed. The span of time for each set of tasks corresponds to the 45 minutes of preparation time. Each preparation period ends at the assigned launch time for that particular group of tasks.



Conflict Sets: C1 = {Tasks 1-10, 11-13}
C2 = {Tasks 11-13, 14-23}
C3 = {Tasks 14-23, 24-26}

Figure 10: Conflict Sets. Tasks 1 through 10 are all assigned a launch time of 1100. Tasks 11 through 13 are to be launched at 1130, tasks 14 through 23 at 1200, and tasks 24 through 26 at 1230. Only one missile at a time may be prepared for launch in any one half-module. Using an assumed 45 minute preparation time for each missile prior to launch, this results in three sets of "conflicting" tasks; only one missile per conflict set may be assigned to any half-module. [After Naval Surface Warfare Center Dahlgren Division, 1997]

Three of the eight task groups require CIII missiles; all other task groups allow either CII or CIII missiles to be used. For the tasks requiring CIII missiles, the M^3 list contains a single entry, assigning CIII missiles number 1 on the list. For the tasks allowing a choice of missiles, CII's (the least capable missile) are given M^3 list position 1, and CIII's position 2. The M^3 list assignment is shown in Table 3.

TASK GROUP	M^3 List	
	CII	CIII
Tasks 1-13	1	2
Tasks 14-26	1	2
Tasks 27-39	N/A	1
Tasks 40-52	N/A	1
Tasks 53-65	1	2
Tasks 66-78	1	2
Tasks 79-91	1	2
Tasks 92-104	N/A	1

Table 3: M^3 List. Three task groups require CIII missiles; all other task groups allow either CII or CIII missiles. Where allowed, the less capable CII missiles are assigned M^3 list position one, and CIII missiles position two.

Scenario missile load-outs are identical for each ship of a particular class, and only two variants of TLAM are included. The cruisers are loaded with 32 TLAM's as follows: 16 CII missiles, with one in Cell 1 of each of the 16 modules, and 16 CIII's, one in Cell 5 of each module. ARLEIGH BURKE class destroyers are assigned 24 TLAM's, 12 CII's, loaded in Cell 1 of each module, and 12 CIII's, loaded in Cell 5 of each module. SPRUANCE class destroyers carry a full load of 61 TLAM's. CII missiles are loaded into Cells 3,4,5, and 6; CIII's are assigned to Cells 1,2,7, and 8. Figure 11 shows a representative load plan for one VLS launcher.

Sample CG Load-out										
Mod	▨	▨	▨	CIII		▨	▨	▨	CII	Mod
2	CII	▨	▨	▨		CIII	▨	▨	▨	1
Mod	▨	▨	▨	CIII		▨	▨	▨	CII	Mod
4	CII	▨	▨	▨		CIII	▨	▨	▨	3
Mod	▨	▨	▨	CIII		▨	▨	▨	CII	Mod
6	CII	▨	▨	▨		CIII	VLS Crane			5
Mod	▨	▨	▨	CIII		▨	▨	▨	CII	Mod
8	CII	▨	▨	▨		CIII	▨	▨	▨	7

Figure 11: Sample Cruiser Load Plan. All ships of a particular class are assigned identical load-outs. This figure shows an example of the load plan for one launcher of a TICONDEROGA class cruiser. Cruisers were loaded with a total of 16 CII missiles and 16 CIII missiles.

B. RESULTS

Table 4 gives a summary of scenario results. Results are provided for the weighted objective function, hierarchical restriction, and heuristic implementations where appropriate. The objective function values for each term are shown, as well as the total solution time. If an operator-selected option, such as primary missile spreading, or designating an expend platform, is not used for a scenario, then "N/A" is recorded for that objective function term.

SCENARIO	OBJECTIVE FUNCTION VALUE BY SUB-PROBLEM								SOLN TIME (SEC)
	1	2	3	4	5	6	7 P/RS/BU	8	
1 Weighted	0	0	N/A	34	3	N/A	13/12/12	0	12
1 HR	0	0	N/A	34	3	N/A	13/8/8	160	26
1 Heuristic	0	0	N/A	34	3	N/A	13/8/8	159	8
1A Weighted	0	N/A	10	8.67	N/A	N/A	13/9/13	0	30
1A HR	0	N/A	10	8.67	N/A	N/A	13/8/8	160	34
1A Heuristic	0	N/A	10	8.67	N/A	N/A	13/8/8	159	15
1B Weighted	0	N/A	0	8.67	N/A	4	13/12/11	0	29
1B HR	0	N/A	10	8.67	N/A	4	13/8/8	160	36
1B Heuristic	0	N/A	10	8.67	N/A	2	13/8/8	159	12
2 Weighted	0	0	N/A	23	3	N/A	26/27/26	0	1176
2 HR	0	0	N/A	23	3	N/A	26/16/16	157	366
2 Heuristic	0	0	N/A	23	3	N/A	26/16/16	156	114
2A Weighted	0	1	N/A	42.5	2	N/A	26/26/26	0	2716
2A HR	0	1	N/A	42.5	2	N/A	26/20/16	146	103
2A Heuristic	0	1	N/A	42.5	2	N/A	26/19/19	146	22
3 HR	0	0	N/A	15.5	3	N/A	52/32/32	137	1625
3 Heuristic	0	0	N/A	15.5	3	N/A	52/32/32	129	1176
4 HR	0	1	N/A	0	4	N/A	65/52/40	127	6133
4 Heuristic	0	1	N/A	0	4	N/A	65/52/48	120	3983
5 HR	0	0	N/A	11.4	4	N/A	78/50/48	268	10976
5 Heuristic	0	0	N/A	11.4	4	N/A	78/50/48	258	2449
5A HR	0	0	65	34	6	N/A	78/52/48	242	7356
5A Heuristic	0	0	65	34	5	N/A	78/52/48	229	1800
5B HR	0	1	N/A	21.4	4	N/A	78/62/48	268	34340
5B Heuristic	0	1	N/A	21.4	4	N/A	78/62/48	260	6488
6 HR	0	0	69	24	6	N/A	124/84/80	209	261
6 Heuristic	0	0	69	24	6	N/A	126/84/78	204	98

Table 4: Scenario Results. The first two base scenarios and their associated modifications have been solved using the weighted objective function method; all scenarios are solved with the hierarchical restriction and heuristic implementations. Objective function values are given for each sub-problem. For sub-problem 7, which minimizes the sum of M^3 list positions, results are given as the sum of M^3 list positions for primary, ready-spare, and back-up task parts. See Table 1 for the correspondence between objective function term and sub-problem number.

The weighted objective function method solution for sub-problem 7 is always poorer than either the hierarchical restriction or heuristic methods; this is an artifact of positive integrality gaps discussed in Chapter II. For the same reason, sub-problem 8, which maximizes residual salvo capability, is never considered with the weighted objective function method. A more detailed discussion of the positive integrality gaps demonstrated by this is presented later in this chapter. The weighted objective function method exhibits rapid growth of solution times with increasing problem size; in fact, this method did not return a solution for any scenario larger than Scenario 2.

We conclude that the weighted objective function is intractable for this application.

A comparison of the solution times and objective function values between the hierarchical restriction and heuristic implementations provides insight into the relative merits of the heuristic. The scenarios presented here yield solution times for the heuristic that are always less than those obtained with the hierarchical restriction method. The absolute difference in solution times becomes more pronounced as the scenarios become more difficult to solve; this is discussed later in this chapter. The heuristic returns the same objective function value as the hierarchical restriction method for the first four sub-problems of every scenario tested. A discrepancy between the heuristic and hierarchical restriction objective function values is not realized until sub-problem 5, which is not optimized with the heuristic implementation, but rather evaluated using the fixed missile allocation from sub-problem 4. Of the 11 scenarios tested, only one, Scenario 5A, shows a difference between the objective function values. The hierarchical restriction yields a superior solution for the only scenario that included back-up spreading. In this case, the heuristic also fixes the allocations to compute the objective function value. Only three scenarios (2A, 4, and 6) show a difference between the two methods for sub-problem 7, which minimizes the sum of M^3 list positions. These are particularly restrictive cases, in which additional considerations make it more challenging for the solver to find a solution within the specified optimality criteria; Scenario 2A divides four available firing platforms between two launch areas, Scenario 4 contains the most tasks of the scenarios with only four firing platforms, and Scenario 6 requires allocations made in Scenario 5 to be fixed while new tasking is added. The solution returned by the heuristic for these more difficult cases illustrates that further testing is required to determine the most robust way of considering M^3 list position. Except in one case, the heuristic method always returns a poorer solution to sub-problem 8, which maximizes residual salvo capability; however, the worst case is within 5% of the hierarchical restriction solution.

The number of variables for each scenario is given in Table 5 for each sub-problem for the hierarchical restriction and the heuristic. The weighted objective function method is solved as a monolithic problem; the number of variables for the entire model is given in the first column, where applicable. The number of discrete variables is reported rather than the total number of variables; this prevents variables whose values have been fixed in a previous sub-problem from being included in the number shown for the heuristic, and is done to allow a meaningful comparison between the relative problem sizes of the hierarchical restriction and heuristic implementations. The total number of variables in each scenario is within 10% of the number of discrete variables. Table 5 highlights the reduction in problem size that the heuristic realizes by evaluating a fixed allocation rather than optimizing sub-problems 5, 6, and 8. Additionally, Scenarios 1, 1A, 1B, 2, 5, and 6 demonstrate the size of problem that can be omitted by not considering sub-problem 2, where possible.

SCENARIO	NUMBER OF DISCRETE VARIABLES BY SUB-PROBLEM							
	1	2	3	4	5	6	7	8
1 Weighted	20519							
1 HR	4325	4325	4322	4325	4329	4329	4325	4837
1 Heuristic	3396	N/A	N/A	4325	8	8	4325	2388
1A Weighted	20519							
1A HR	4325	4325	4322	4325	4329	4329	4325	4837
1A Heuristic	4325	N/A	4321	4324	7	7	4324	2387
1B Weighted	20519							
1B HR	4325	4325	4322	4325	4329	4329	4325	4837
1B Heuristic	4325	N/A	4321	4324	7	7	4324	2387
2 Weighted	40526							
2 HR	8650	8650	8643	8646	8650	8650	8646	9158
2 Heuristic	6789	N/A	N/A	8646	8	8	8646	260
2A Weighted	40526							
2A HR	8650	8650	8643	8646	8650	8650	8646	9158
2A Heuristic	6792	8646	N/A	8646	8	8	8646	260
3 HR	12938	12938	12935	12938	12942	12942	12938	15400
3 Heuristic	10153	12938	N/A	12938	8	8	12938	260
4 HR	17259	17259	17256	17259	17263	17263	17259	19721
4 Heuristic	13546	17259	N/A	17259	8	8	17259	260
5 HR	38512	38512	38512	38512	38512	38512	38512	42444
5 Heuristic	33878	N/A	N/A	38519	14	14	38519	455
5A HR	38512	38512	38512	38512	38512	38512	38512	42444
5A Heuristic	33878	N/A	38512	38516	11	11	38516	452
5B HR	38512	38512	38512	38512	38512	38512	38512	42444
5B Heuristic	38519	38519	N/A	38519	14	14	38522	455
6 HR	53766	53766	53766	53766	53766	53766	53766	57698
6 Heuristic	47276	N/A	53766	53770	11	11	53770	452

Table 5: Number of Discrete Variables by Sub-Problem. For the scenarios implemented with weighted objective functions, the number of discrete variables for the entire model is shown in the first column. The number of discrete variables is shown rather than the total number of variables. This prevents counting variables whose values have been fixed under the heuristic method. The total number of variables for each scenario is within 10% of the number of discrete variables for each method.

Table 6 gives the number of constraints for each sub-problem for the hierarchical restriction and the heuristic; the total number of constraints for the weighted objective function models is shown in the first column, where appropriate. Note there is no appreciable difference in the reported number of constraints between the hierarchical restriction and heuristic models; however, in actuality, with the heuristic method many of these equations are reduced to expressing that a constant is equal to a constant, which implies that the number of coefficients in the constraint matrix and/or the total number of "real" constraints is greatly reduced for the heuristic implementation.

SCENARIO	NUMBER OF CONSTRAINTS BY SUB-PROBLEM							
	1	2	3	4	5	6	7	8
1 Weighted	56412							
1 HR	4377	4377	4378	4391	4399	4407	4407	5050
1 Heuristic	3320	N/A	N/A	4391	4399	4403	3459	4339
1A Weighted	56412							
1A HR	4377	4377	4378	4388	4396	4404	4404	5047
1A Heuristic	4325	N/A	4377	4388	4396	4400	4336	4783
1B Weighted	56412							
1B HR	4377	4377	4378	4388	4396	4404	4404	5047
1B Heuristic	4325	N/A	4377	4388	4396	4400	4336	4783
2 Weighted	111762							
2 HR	8708	8708	8709	8722	8730	8738	8738	9381
2 Heuristic	6954	N/A	N/A	8722	8730	8734	8730	9117
2A Weighted	138386							
2A HR	5479	5479	5480	5493	5501	5509	5509	6152
2A Heuristic	5427	5479	N/A	5489	5497	5501	5437	5884
3 HR	13365	13365	13366	13379	13387	13395	13395	14038
3 Heuristic	13365	13365	N/A	13379	13387	13391	13390	13774
4 HR	17696	17696	17697	17710	17718	17726	17726	18369
4 Heuristic	17696	17696	N/A	17710	17718	17722	17721	18105
5 HR	35340	35340	35340	35363	35377	35384	35377	36052
5 Heuristic	30477	N/A	N/A	35363	35377	35384	35380	36052
5A HR	35340	35340	35340	35354	35368	35375	35368	36043
5A Heuristic	30477	N/A	35340	35354	35368	35375	35371	36043
5B HR	35340	35340	35340	35363	35377	35384	35377	36052
5B Heuristic	34846	35340	N/A	35363	35377	35384	35380	36052
6 HR	53745	53745	53745	53759	53773	53780	53773	54448
6 Heuristic	45953	N/A	53745	53759	53773	53780	53776	54448

Table 6: Number of Constraints by Sub-Problem. For the scenarios implemented with weighted objective functions, the number of constraints for the entire model is shown in the first column. There is no appreciable difference in the number of equations reported for the hierarchical restriction and heuristic methods; however, in actuality, many of these constraints may be reduced to stating that a constant is equal to a constant. This leads to substantial reduction in the number of "real" constraints that must be considered with the heuristic method.

Table 7 shows the relative integrality gap achieved for each sub-problem. A maximum allowed relative gap of 5% was used with the weighted objective function. This results in absolute gaps on the order of 5.6×10^{11}. As discussed in Chapter II, the effect of this gap is seen in the sub-optimal values for variables contained in the lower priority objective function terms. The model selects missiles that result in a sub-optimal solution for considerations such as the sum of M^3 positions and residual salvo capability; this is further indication that a weighted objective function is not well suited for the predesignation application. For the hierarchical restriction method, we observe that, for some sub-problems, almost all of the solution time is spent trying to find a feasible solution. Once this initial solution is found, an optimal solution is returned in a matter of seconds. This encourages us to experiment with an allowed relative gap of 100%, with the intention of having the branch and bound process terminate as soon as a feasible solution is found. We find that, for most scenarios, this results in negligible returned integrality gaps; based on this we do not believe that solution times would be adversely affected by specifying a (reasonable) tolerance on the order of 5%. This has not been done because, in almost all cases, a tolerance of 100% resulted in returned gaps of less than 5%. (The exceptions were Scenario 5B, sub-problem 2, and Scenario 6, sub-problem 1; we set the tolerance at 10%.) An integrality gap greater than zero only occurs in some instances of sub-problem 4, which minimizes the difference between the number of missiles remaining on a firing platform and the mean number of missiles on all firing platforms. Even in this case, the largest gap we observe is 5%; this equates to a difference of one missile. Note that because the hierarchical restriction and heuristic both solve sub-problem 4 in its entirety, the integrality gaps and objective function values for this sub-problem are identical for both methods for all scenarios.

SCENARIO	RETURNED RELATIVE INTEGRALITY GAP BY SUB-PROBLEM							
	1	2	3	4	5	6	7	8
1 Weighted	0.03							
1 HR	0	0	0	0.03	0	0	0	0
1 Heuristic	0	0	0	0.03	0	0	0	0
1A Weighted	0.05							
1A HR	0	0	0	0	0	0	0	0
1A Heuristic	0	0	0	0	0	0	0	0
1B Weighted	0.05							
1B HR	0	0	0	0	0	0	0	0
1B Heuristic	0	0	0	0	0	0	0	0
2 Weighted	0.05							
2 HR	0	0	0	0.04	0	0	0	0
2 Heuristic	0	0	0	0.04	0	0	0	0
2A Weighted	0.05							
2A HR	0	0	0	0	0	0	0	0
2A Heuristic	0	0	0	0	0	0	0	0
3 HR	0	0	0	0	0	0	0	0
3 Heuristic	0	0	0	0	0	0	0	0
4 HR	0	0	0	0	0	0	0	0
4 Heuristic	0	0	0	0	0	0	0	0
5 HR	0	0	0	0	0	0	0	0
5 Heuristic	0	0	0	0	0	0	0	0
5A HR	0	0	0	0.03	0	0	0	0
5A Heuristic	0	0	0	0.03	0	0	0	0
5B HR	0	0	0	0.04	0	0	0	0
5B Heuristic	0	0	0	0.04	0	0	0	0
6 HR	0	0	0	0	0	0	0	0
6 Heuristic	0	0	0	0	0	0	0	0

Table 7: Returned Relative Integrality Gap by Sub-Problem. All gaps are no more than 0.05 (i.e., 5%).

Solution times for each sub-problem are reported in Table 8. It is apparent that the heuristic takes longer to solve sub-problem 1 than the hierarchical restriction; this is because the heuristic attempts to first meet all allocations without using any platforms with employment penalties. If the required tasking cannot be met with this restricted set of available platforms, the sub-problem is re-solved using all of the firing platforms.

Additionally, the heuristic will attempt to solve the sub-problem for separate task parts, fixing the allocations after each task part is solved for. This may not allow a feasible solution, in which case the sub-problem is re-solved for all task parts together. In practice, this can lead to multiple attempts to solve sub-problem 1: (i) by individual task parts without considering firing platforms with employment penalties, (ii) for all task parts simultaneously, with the restricted set of platforms, (iii) by individual task parts with all firing platforms considered, and finally (iv) for all task parts simultaneously, considering all firing platforms. However, the heuristic does reduce the total solve times in the instances where it omits sub-problem 2. Scenarios that do not designate expend platforms permit the elimination of sub-problem 3, reducing solve time. The full hierarchical restriction for sub-problem 4 is solved by the heuristic, and the solution times for the two methods are, in general, comparable. Evaluating sub-problems 5, 6, and 8 using a fixed allocation provides a significant reduction in solution time. The heuristic attempts to solve sub-problem 7 for individual task parts, as it does for sub-problem 1. For the scenarios here, the results are inconclusive. In instances where a feasible solution can be found by solving for primary, ready-spare, and back-up task parts separately and fixing allocations after each task part, the heuristic takes less time. In cases where the heuristic cannot find a feasible solution, the entire sub-problem is re-solved for all task parts together, and the solution time for sub-problem 7 is greater than for the hierarchical restriction implementation.

Unacceptably high solution times (e.g., about 44,000 seconds for one sub-problem instance) encouraged us to explore alternate executions of the CPLEX branch-and-bound algorithm. To this end, care has been taken to achieve the best possible performance from CPLEX for all executions of the monolithic, hierarchical restriction, and heuristic methods. Extensive testing has determined the CPLEX parameter settings that result in the fastest execution of the branch-and-bound algorithm for each set of problem instances. For all monolithic problems, and for (both the hierarchical restriction and the heuristic implementation of) sub-problems 1 through 6, the settings that yield the best average performance are: (i) select the branching variable based on "pseudo reduced costs", (ii) solve the initial LP relaxation with the primal Simplex method, (iii) solve the sub-problems at each node with the primal Simplex method, (iv) choose the next node in the branch-and-bound tree for processing based on the best estimate value (given an integer completion) of the integer objective function, and (v) turn off the intrinsic CPLEX heuristic to realize feasible integer completions. Scenario 5B, because of the three ship employment penalties, is more difficult to solve than the corresponding sub-problem for the other scenarios. In this instance, the best settings are: (i) branch upward first at each node, (ii) select the branch variable based on solving a sample of sub-problems to determine which branching variable might yield the best results, and (iii) use probing to tighten bounds at the root node (e.g., by fixing certain variable values). The best settings for sub-problem 7 include all of the above settings, but rather than selecting the branching variable based on "pseudo reduced costs", the selection is based on "pseudo costs". The best settings for sub-problem 8 include all parameter settings used for the first six sub-problems, and the branching and probing strategies used to solve sub-problem 7. [CPLEX, 1995]

SCENARIO	SOLUTION TIME BY SUB-PROBLEM (SECONDS) ON A 333 MHz PC WITH A PENTIUM II PROCESSOR							
	1	2	3	4	5	6	7	8
1 Weighted	12							
1 HR	0.7	0.9	0.8	4.6	0.8	0.7	1.0	16.5
1 Heuristic	0.9	N/A	N/A	4.7	0.4	0.4	1.4	0.4
1A Weighted	30							
1A HR	0.9	0.7	2.8	0.7	0.7	0.7	3.4	24.2
1A Heuristic	4.3	N/A	7.7	0.8	0.4	0.3	0.6	0.4
1B Weighted	29							
1B HR	0.7	0.7	2.5	2.1	1.4	0.7	2.1	26.3
1B Heuristic	1.2	N/A	7.8	0.8	0.4	0.3	1.4	0.4
2 Weighted	1176							
2 HR	1.5	1.5	6.7	3.8	6.2	5.2	259.0	82.4
2 Heuristic	1.9	N/A	N/A	3.5	0.7	0.8	106.5	0.7
2A Weighted	2716							
2A HR	1.1	5.8	1.5	1.7	1.4	1.4	10.0	80.5
2A Heuristic	9.9	6.2	N/A	1.7	0.5	0.6	2.6	0.6
3 HR	8.6	375.7	55.6	64.0	76.2	57.3	669.9	317.6
3 Heuristic	18.1	379.0	N/A	63.0	1.9	1.4	710.5	1.8
4 HR	48.7	662.6	256.6	317.9	292.6	386.2	2813.9	1354.2
4 Heuristic	60.2	634.1	N/A	317.4	1.3	1.3	2967.0	2.0
5 HR	13.0	45.0	47.4	731.0	465.8	1301.9	3522.3	4849.6
5 Heuristic	17.1	N/A	N/A	215.8	2.9	3.5	2206.1	3.5
5A HR	11.4	45.0	396.9	617.6	78.3	193.4	1211.9	4801.3
5A Heuristic	17.8	N/A	397.3	123.8	2.9	2.8	1252.7	2.8
5B HR	11.3	23495.2	171.3	915.2	469.5	855.8	4330.0	4091.6
5B Heuristic	23.1	1487.3	N/A	1127.3	3.1	2.9	3840.9	3.5
6 HR	8.4	121.9	8.1	14.2	10.5	10.6	60.8	27.0
6 Heuristic	19.83	N/A	11.6	8.0	4.3	4.6	45.4	4.6

Table 8: Solution Time by Sub-Problem. The heuristic results in longer solve times for sub-problem 1; the benefit of the heuristic is realized where the method of solving sub-problem 1 allows sub-problem 2 to be omitted. A further reduction is realized by the heuristic in cases that do not require an expend platform, allowing sub-problem 3 to be excluded. Evaluating sub-problems 5, 6, and 8 using a fixed allocation produces the greatest difference in total solution times between the hierarchical restriction and the heuristic. Attempting to solve sub-problem 7 for individual task parts with the heuristic produces inconclusive results. In cases where a feasible solution can be found by solving for primary, ready-spare, and back-up parts in succession, the heuristic is faster. If a feasible solution cannot be found using this method, then the sub-problem is re-solved for all task parts together, and the hierarchical restriction is faster than the heuristic.

IV. CONCLUSIONS

A. GENERAL

Mixed integer programming does not lend itself readily to Tactical Decision Aids in general, and the TSC predesignation application in particular. There is no guarantee that a mixed integer program that has performed well for certain problem instances will perform adequately for alternate scenarios. The time required to reach a solution can vary widely as input data changes, and integer programming solvers (with tailored parameter settings) that appear adequate for one problem instance may prove disastrous when applied to another. These concerns complicate adoption of mixed integer programming as the underpinning of a tool that is intended for use by an operator in the fleet, who requires a useable solution in a reasonable amount of time to execute National Command Authority tasking. Through our implementation methods, we have attempted to provide the best possible structure for the use of an imperfect tool; it must be noted that each method has significant shortcomings that must be considered before the form of the TSC predesignation model is finalized.

B. WEIGHTED OBJECTIVE FUNCTION IMPLEMENTATION

The weighted objective function is unsuitable for the TSC application because of the complication induced in integrality gaps and numerical stability. Poor solutions and rapid increase of solution times with problem size are unacceptable. The weights used for our scenarios were computed using assumptions about the maximum number of firing platforms, weapons, and tasks. To use the weighted objective function in a real-world operational context, the weights would have to be modified to reflect the actual force composition and level of tasking for each scenario; the validity of the model would depend on the extent to which the operator could perform these modifications.

C. HIERARCHICAL RESTRICTION AND HEURISTIC

Our hierarchical restriction accurately captures the desired priorities, and produces the best obtainable (although not necessarily optimal) solution to the predesignation problem. However, this method requires long solution times even for moderate-size scenarios. As strike weapons are developed that have preparation times substantially shorter than TLAM, the on-scene decision maker will need a tool that is capable of rapidly returning a solution. Even for the TLAM-only scenarios presented here, the hierarchical restriction does not appear to be sufficiently responsive.

The heuristic returns solutions of reasonable quality in a significantly reduced amount of time for all the scenarios tested. This method also recognizes all priority levels. However, the heuristic presented here is a first step in an evolution to a product robust enough for operational use. There are technical and doctrinal challenges that must be resolved before a heuristic is selected for use in the TSC predesignation system. In worst-case scenarios, the heuristic may need to solve the full hierarchical restriction for a sub-problem after several attempts to solve a restricted version of that sub-problem. This can lead to solution times that are longer for the heuristic than for the seminal hierarchical restriction. Additionally, the manner in which the heuristic evaluates fixed allocations for certain objective functions, rather than optimizing, needs to be clearly presented to the potential users of this product. The ordering of the priorities presented in this thesis has been developed jointly with NSWCDD [Fennemore, 1998c], and informally reviewed by members of Tactical Training Group Atlantic [Williams, 1998]; however, these orderings do not reflect official guidance, and will require doctrinal approval.

D. RECOMMENDATIONS

We recommend that NSWCDD develop a non-integer programming-based heuristic for TSC predesignation. However, a signal disadvantage of such a heuristic is that it is seldom possible to objectively assess the quality of its solutions. The mathematical models and formal methods introduced here may be useful in independently assessing the quality of the non-integer programming-based heuristic solution. Due to the limitations of mixed integer programming and of the heuristic presented here, we further recommend that at an early stage of development, the heuristic be presented to the strike warfare community for review. The goal of this review by decision makers is to determine its suitability for tactical systems, and to determine the correct ordering of priorities. We recommend that NSWCDD advocate the acceptance of some heuristic by the Naval Doctrine Command, Fleet Commanders, and Theater CINC's, and its inclusion in the Naval Warfare Publications and Concepts of Operations that govern the conduct of strike warfare. It is crucial that strike planning be standardized in published doctrine so that it may be carried out with the effectiveness that it deserves.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

CPLEX, 1995, CPLEX Optimization, Inc., *Using the CPLEX Callable Library*, 1995.

Fennemore, C., 1998a, Naval Surface Warfare Dahlgren Division, Dahlgren, VA, interview, 17 November.

Fennemore, C., 1998b, Naval Surface Warfare Dahlgren Division, Dahlgren, VA, interview, 19 November.

Fennemore, C., 1998c, Naval Surface Warfare Dahlgren Division, Dahlgren, VA, interview, 20 November.

Fennemore, C., 1999a, Naval Surface Warfare Center Dahlgren Division, phone conversation, 11 February.

Fennemore, C., 1999b, Naval Surface Warfare Center Dahlgren Division, e-mail, Subject: Strike Predesignation Example Sequence, 18 March 1999.

Fennemore, C., 1999c "Land Attack Predesignation Tasking Proposal", proposal submitted by NSWCDD to Don Wagner, Code 311, Office of Naval Research.

GAMS, 1999, GAMS Version 2.50 Distribution 18.1, GAMS Development Corporation, Washington, DC.

ILOG, 1999, CPLEX Version 6.5, ILOG CPLEX Division, Incline Village, NV.

Kukendall, S.D., 1998, *Optimizing Selection of Tomahawk Cruise Missiles*, Masters Thesis, Naval Postgraduate School, Monterey, CA, March.

Naval Surface Warfare Center Dahlgren Division, 1997, NSWCDD/TR-97/18, *Advanced TOMAHAWK Weapons Control System Predesignation*, James Allewelt and others, February.

Naval Surface Warfare Center Dahlgren Division, 1999, NSWCDD/TR-99/4, *Submarine Predesignation*, C. Fennemore, D. Oliver, R. Taft, March.

Steuer, R., 1986, *Multiple Criteria Optimization: Theory, Computation, and Application*, Wiley.

U.S. Navy, 1994, *Introduction and Description, VLS MK 41 Mods 0/1/2/7*, SW394-AF-MMO-020/VLS Revision 3.

U.S. Navy, 1999a, *Operation Desert Storm*, http://www.chinfo.navy.mil/navpalib/weapons/missiles/tomahawk/tlam 1219.jpg.

U.S. Navy, 1999b, *USS SAN JACINTO (CG-56) Pictures*, http://www.spear.navy.mil/SHIPS/CG56/CG56P.GIF.

U.S. Navy, 1999c, *USS MITSCHER (DDG-57), Welcome Aboard*, http://www.spear.navy.mil/ships/ddg57/.

U.S. Navy, 1999d, *USS JOHN HANCOCK (DD-981) Pictures*, http://www.spear.navy.mil/ships/dd981/.

Williams, C., 1998, Tactical Training Group Atlantic, Dam Neck, VA, interview, 23 November.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
8725 John J. Kingman Rd., STE 0944
Fort Belvoir, Virginia 22060-6218

2. Dudley Knox Library 2
Naval Postgraduate School
411 Dyer Rd.
Monterey, California 93943-5101

3. Professor Alexandra M. Newman 2
Department of Operations Research
Naval Postgraduate School
Monterey, California 93943-5000

4. Professor Gerald G. Brown 2
Department of Operations Research
Naval Postgraduate School
Monterey, California 93943-5000

5. Dahlgren Division 2
Naval Surface Warfare Center
Combat Systems Branch
Attn: Charles Fennemore Code B35
Dahlgren, Virginia 22448-5100

6. LT Brian Kirk 2
737 Brock Drive
Corpus Christi, Texas 78412